CIBERSEGURIDAD

‘Bootcamp IX’

Informe Práctica Blue Team.

Maximiliano Dariel Altamirano.

Academia KeepCoding.

**Enunciado:**

Queremos montar la siguiente infraestructura:



**Los requisitos que debe cumplir son los siguientes:**

1. Debe tener un pfSense en que se interconecten las redes LAN, DMZ y DMZ2

2. En la red LAN debe haber un equipo Windows 11 que envíe logs al servidor de Elastic.

3. En la red DMZ debe haber un honeypot que envíe los logs al servidor de Elastic.

1. Este honeypot no debe tener acceso a ninguna red interna (LAN, DMZ2...) y debe ser

accesible desde el exterior (red WAN) en ambos sentidos.

4. En la red DMZ2 debe haber otra fuente diferente de logs a las dos mencionadas anteriormente.

Se propone Suricata o Apache Server como posibles fuentes, pero se deja a elección del alumno.

5. El servidor de Elastic debe recibir, almacenar y poder visualizar los logs del honeypot, el

Windows 11 y la fuente elegida ubicada en la DMZ2.

**Criterios de evaluación de la memoria:**

1. Debe contener evidencias y explicaciones que demuestren la correcta creación de la

infraestructura de red en el pfSense.

2. Debe contener explicación y captura de las reglas de firewall elegidas para cada red (WAN, NAT,

LAN, DMZ y DMZ2)

3. Debe contener evidencias de las políticas e integraciones asignadas a cada agente del SIEM

(Elastic)

4. Debe contener evidencias que demuestren la correcta recepción de los logs, de todas las fuentes

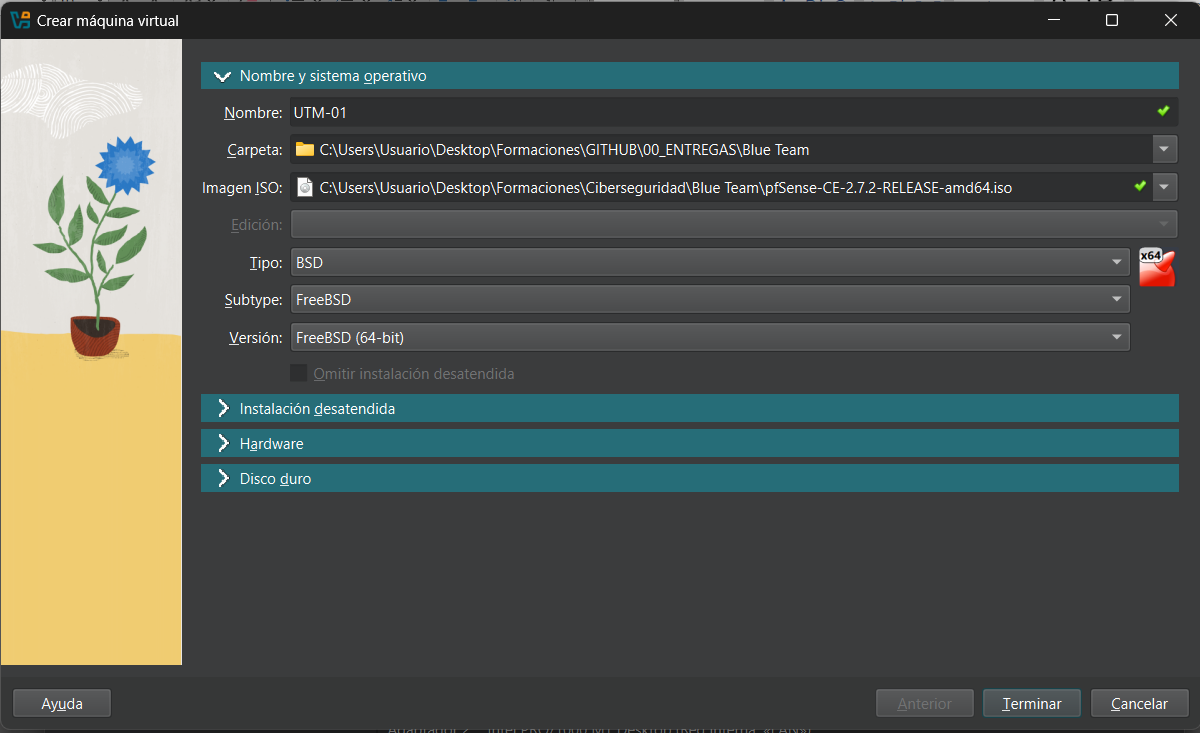
especificadas en el enunciado, en el SIEM (Elastic).

**Desarrollo:**

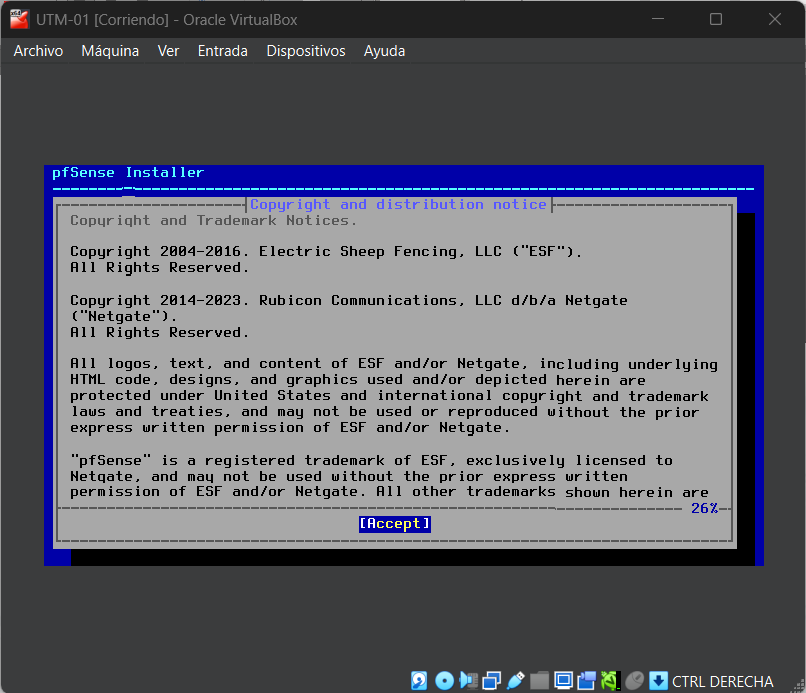
**1. Debe tener un pfSense en que se interconecten las redes LAN, DMZ y DMZ2.**

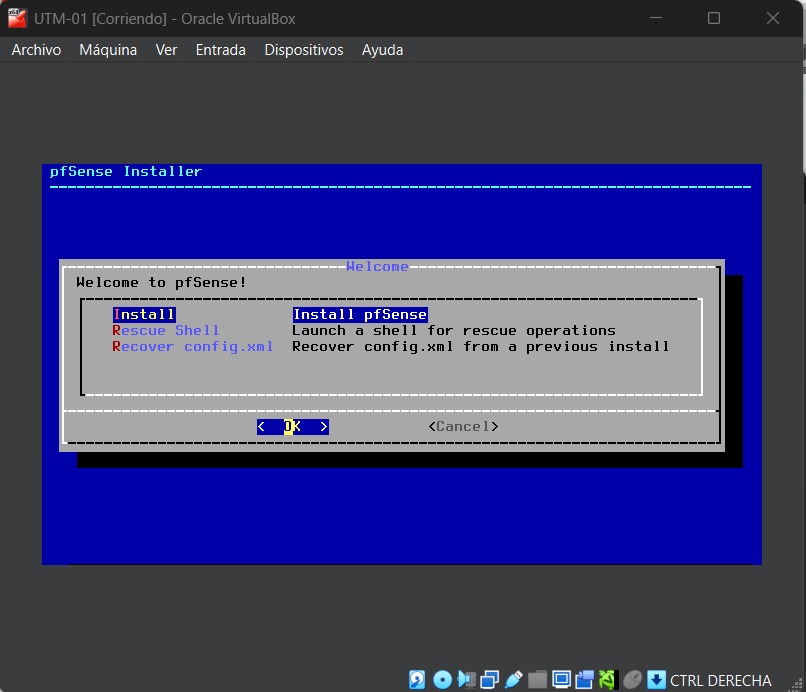
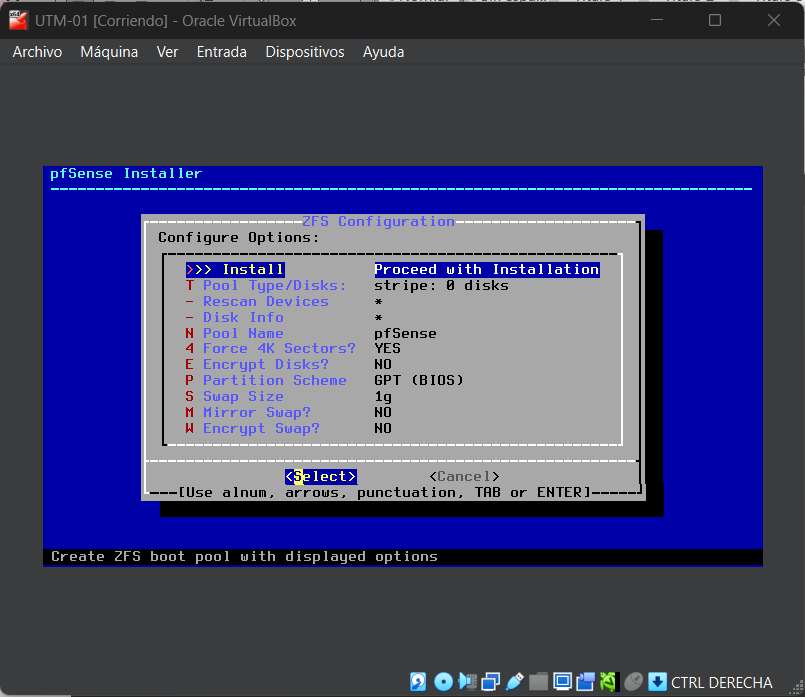
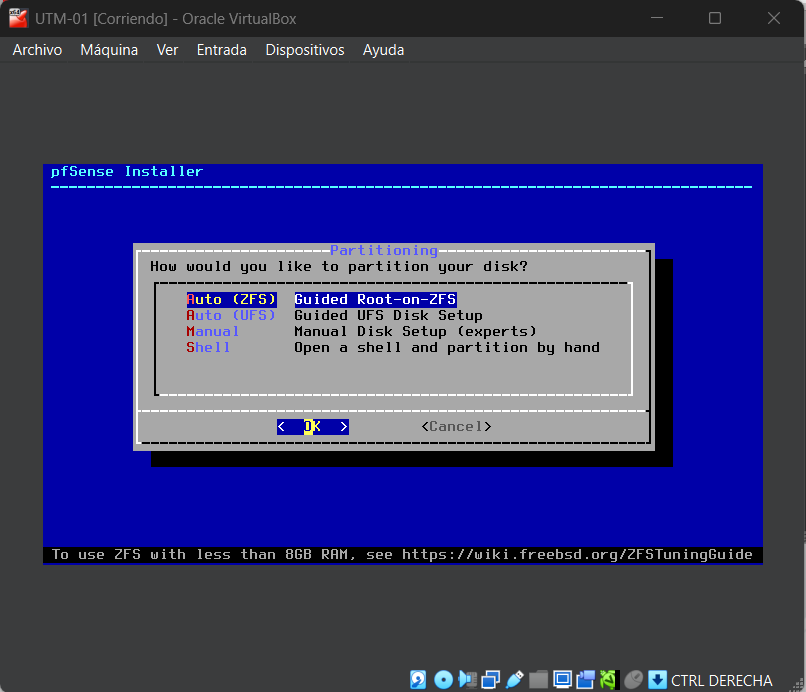
Comenzamos instalando y configurando el pfSense. Previamente deberemos disponer de un virtualizador (VirtualBox) y de la imagen ISO para generar nuestro UTM-01 (pfSense).

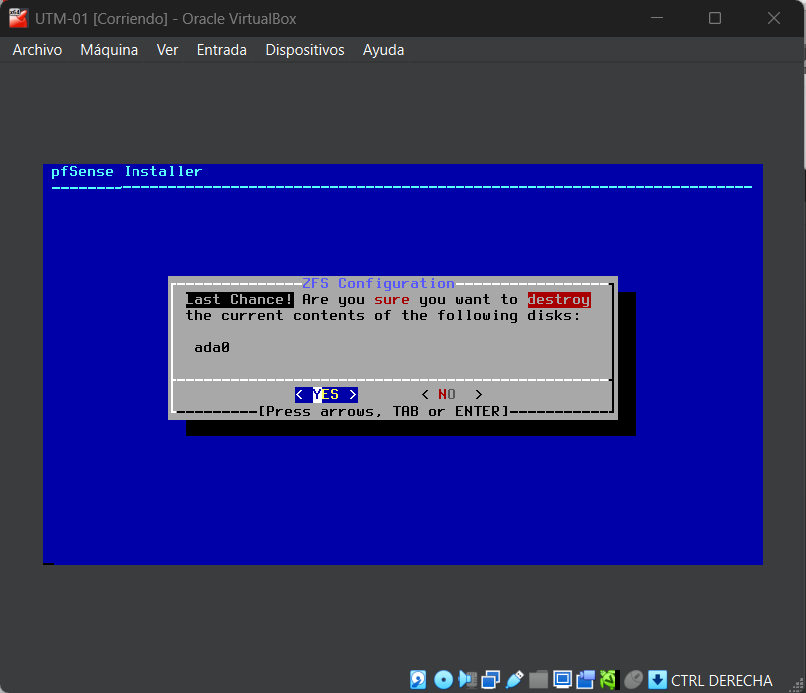
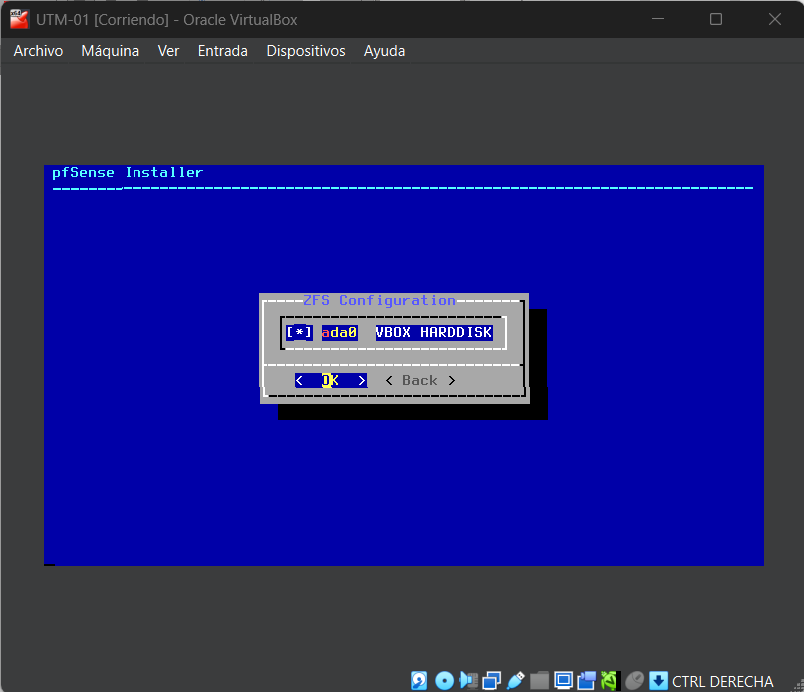
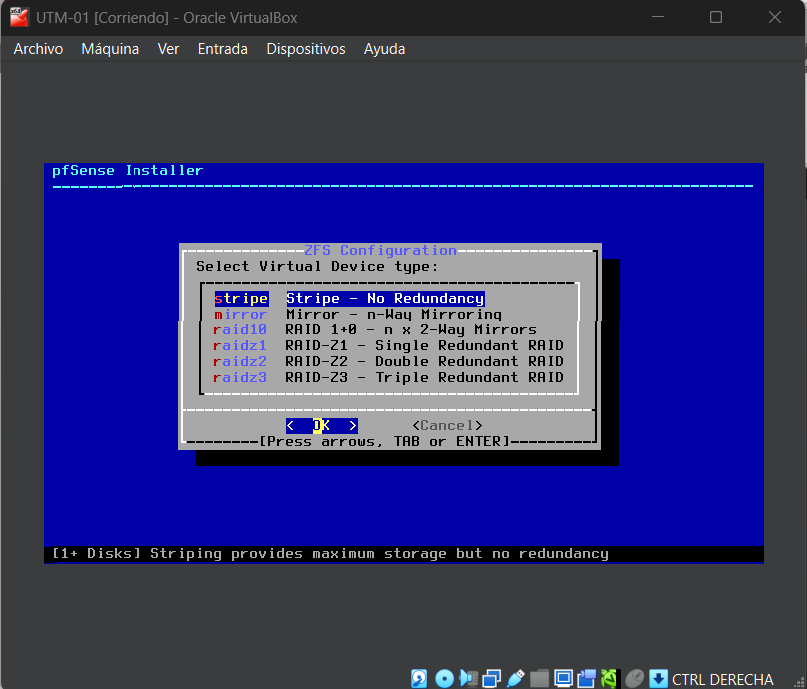
Creamos nuestra máquina Virtual con las siguientes configuraciones:

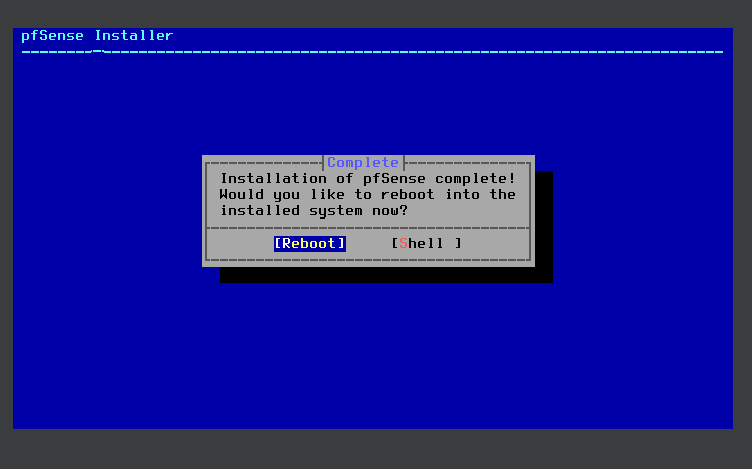
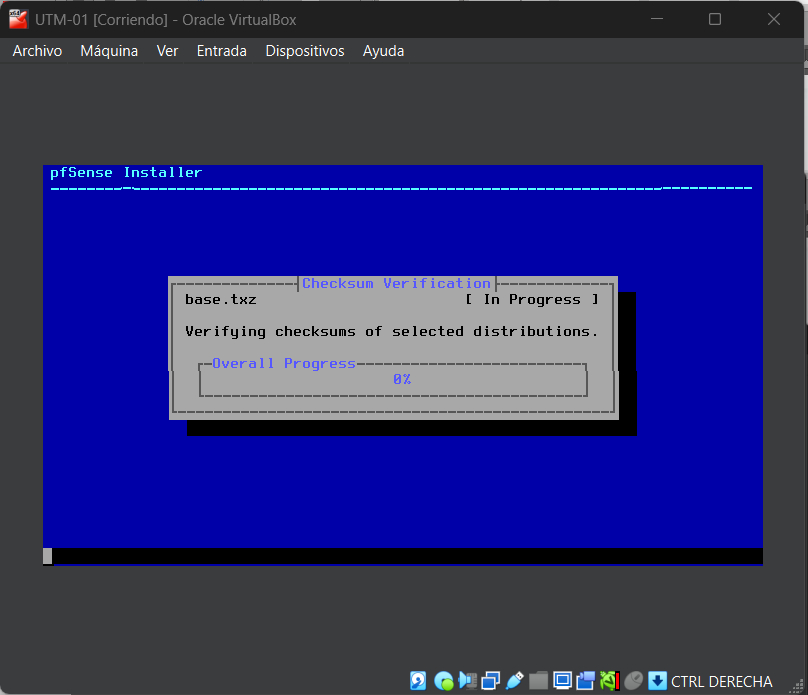
Nombre: UTM-01; Tipo: BSD; Subtype: FreeBSD; Versión: FreeBSD (64-bit)  


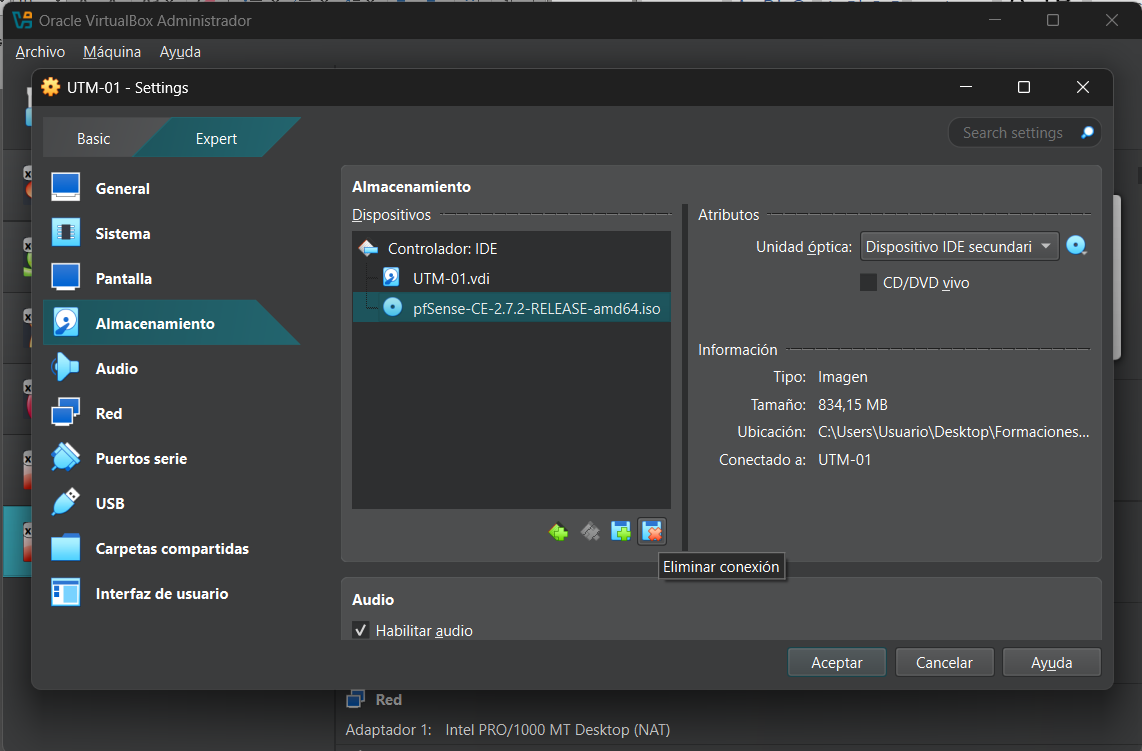
Lanzamos la maquina desde Virtual Box y comenzamos con las configuraciones.

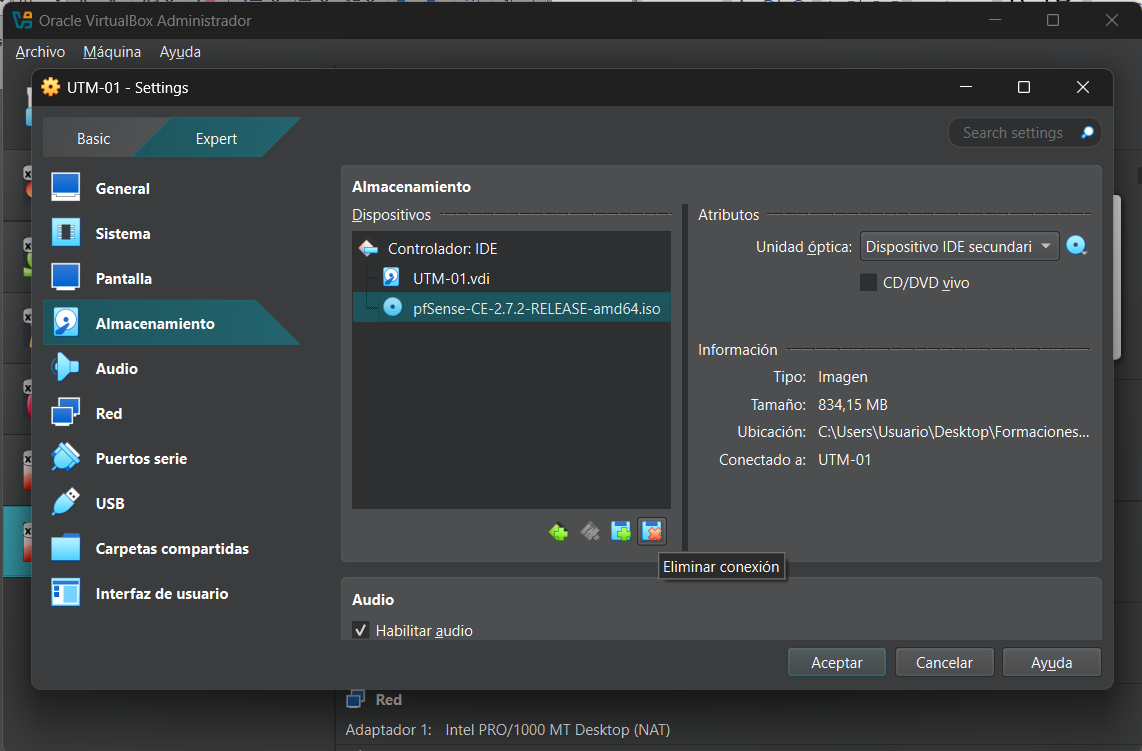






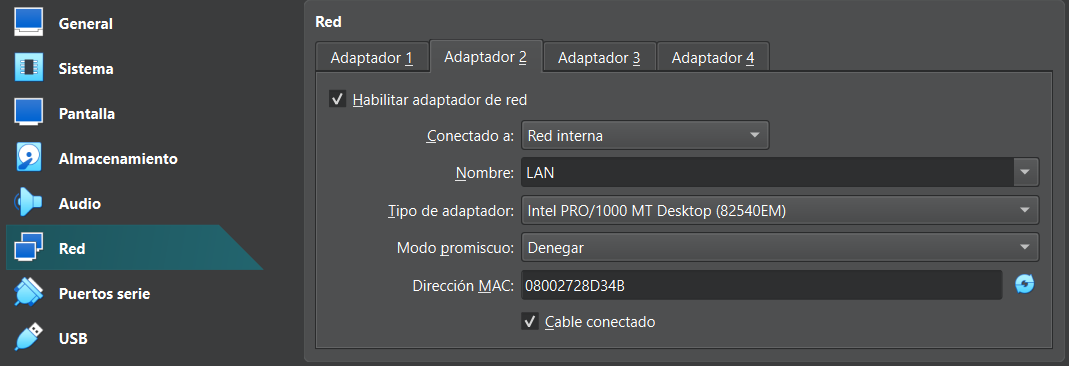


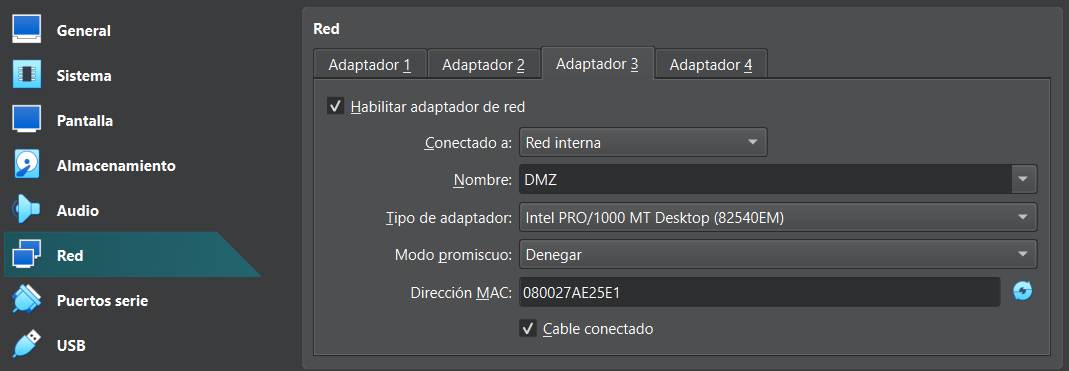
En este punto debemos apagar la máquina virtual y quitar el disco ISO desde las configuraciones de inicio, con este procedimiento evitamos que al lanzar nuevamente nuestro UTM-01 repita la instalación ya que generada.



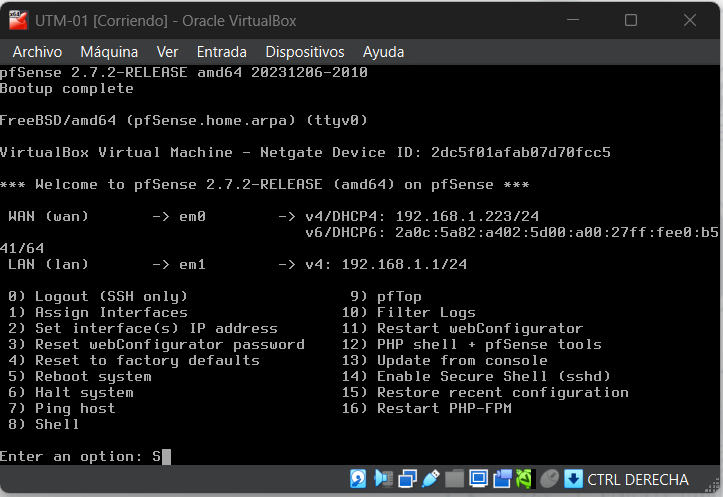
Configuramos los adaptadores renombrando nuestras redes internas, nuevamente desde configuraciones en el apartado Red configuramos los parámetros.

Adaptador 1 | Conectado a: Adaptador puente | Nombre: default

Adaptador 2 | Conectado a: Red interna | Nombre: LAN

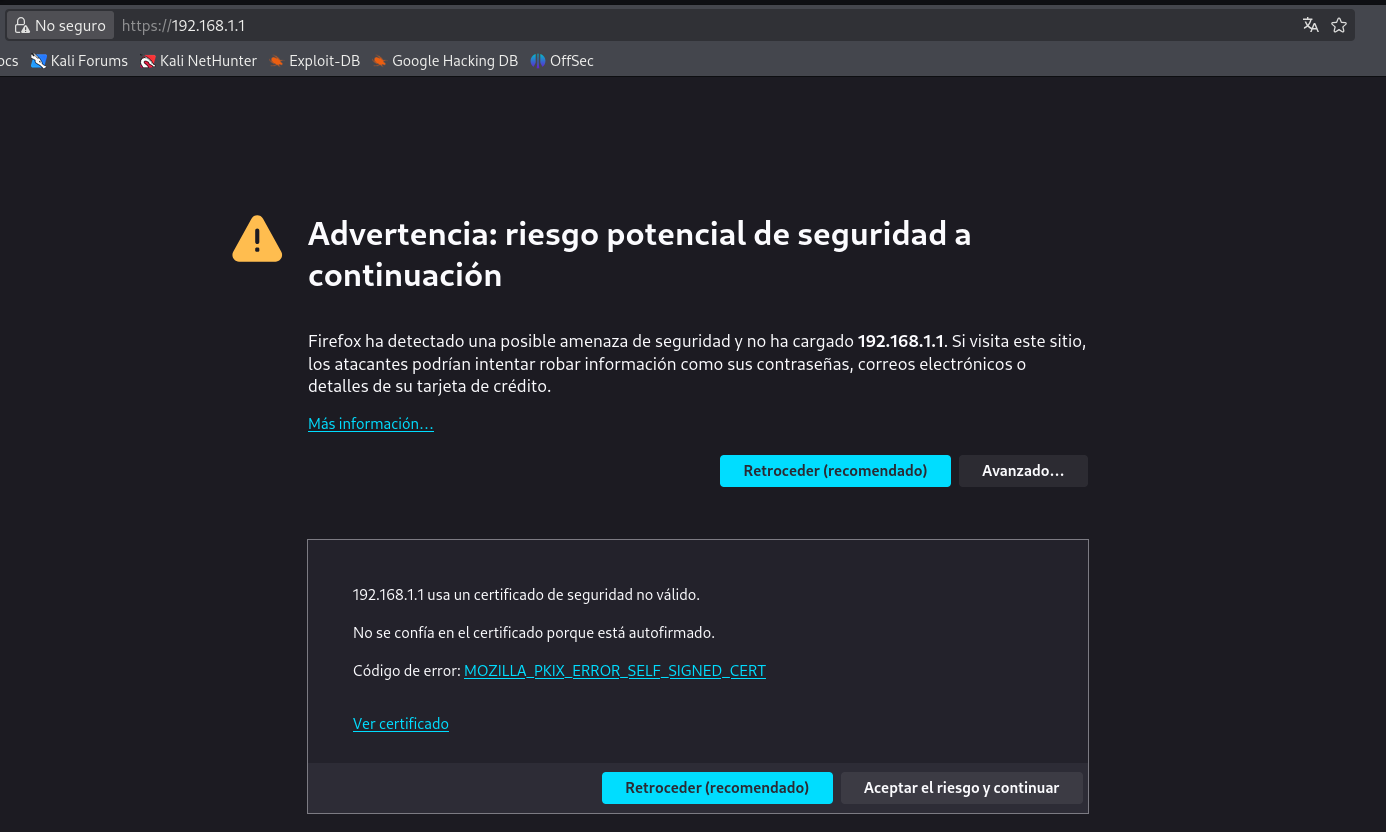
Adaptador 3 | Conectado a: Red interna | Nombre: DMZ

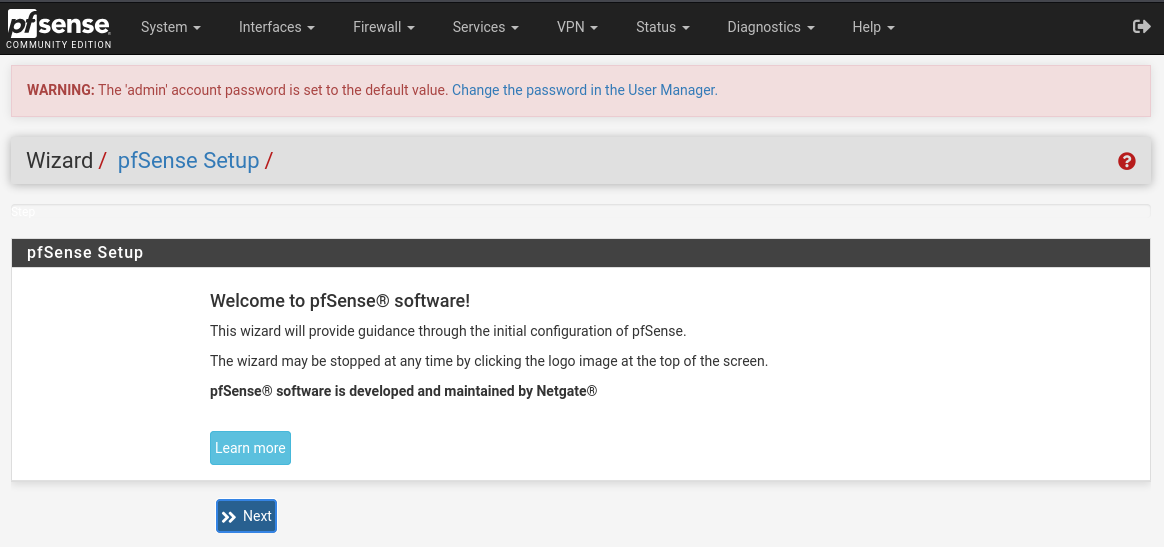
Adaptador 4 | Conectado a: Red interna | Nombre: DMZ2

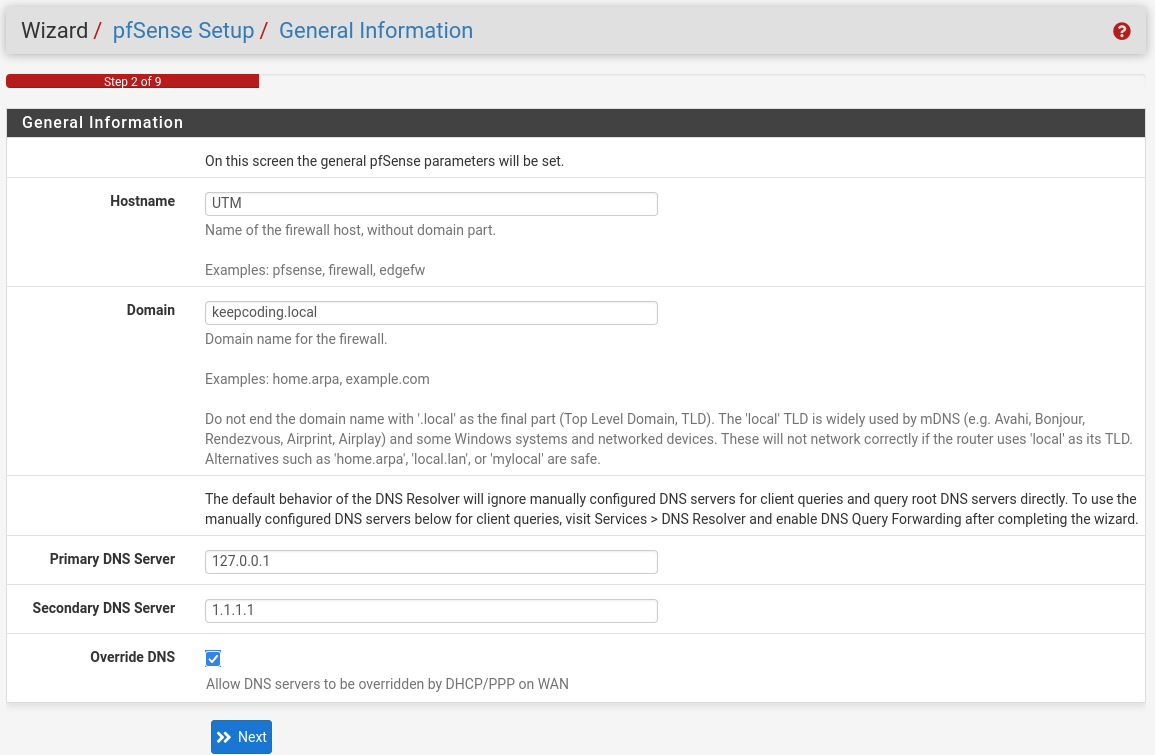
Ya gestionadas nuestras configuraciones iniciales en la interfaz de red, arrancamos nuestra maquina UTM-01 validando las configuraciones realizadas:

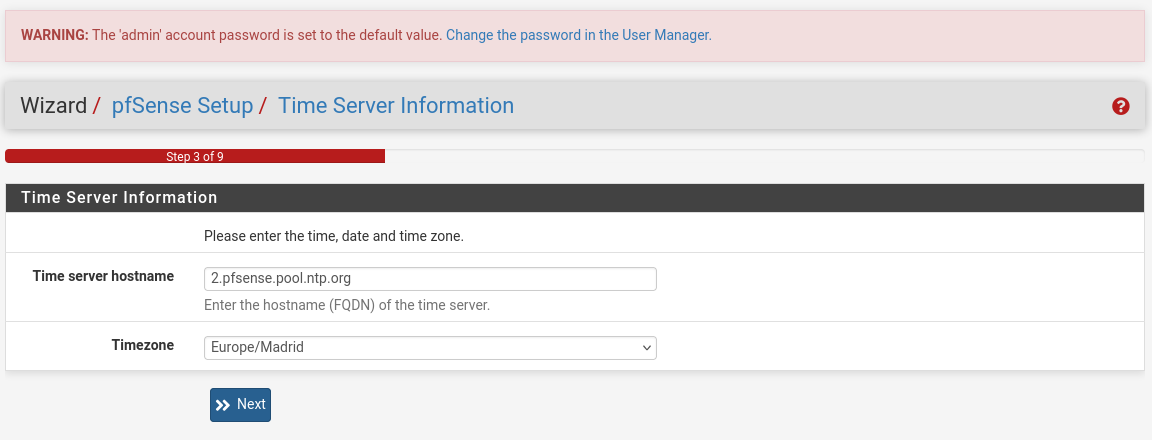
**Configuración pfSense.**  
  
Para realizar las configuraciones de cada red debemos acceder desde un navegador a la dirección IP de nuestra LAN, utilizaremos Kali para gestionarlo.

Nuestra dirección por defecto será 192.168.1.1

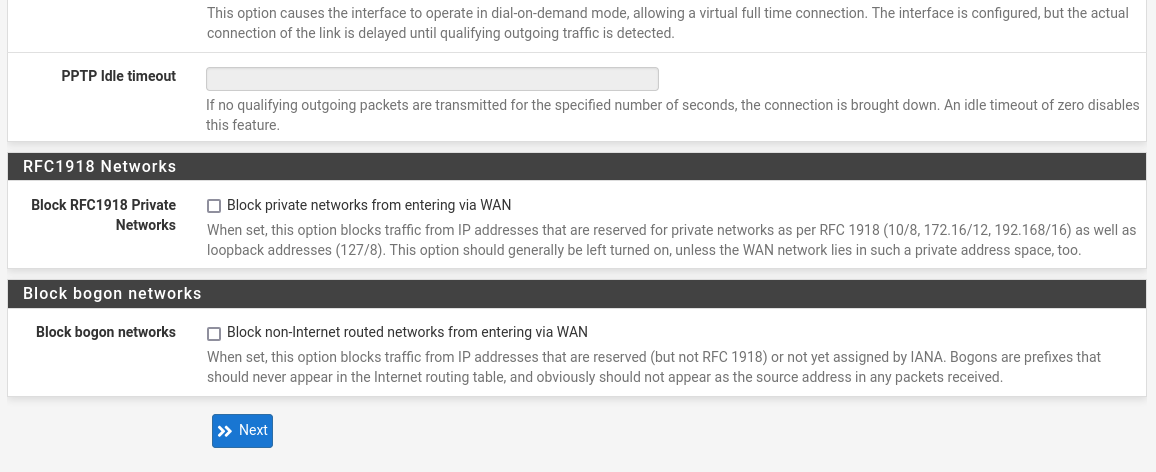
Debemos seleccionar “Aceptar el riesgo y continuar” para acceder al portal.

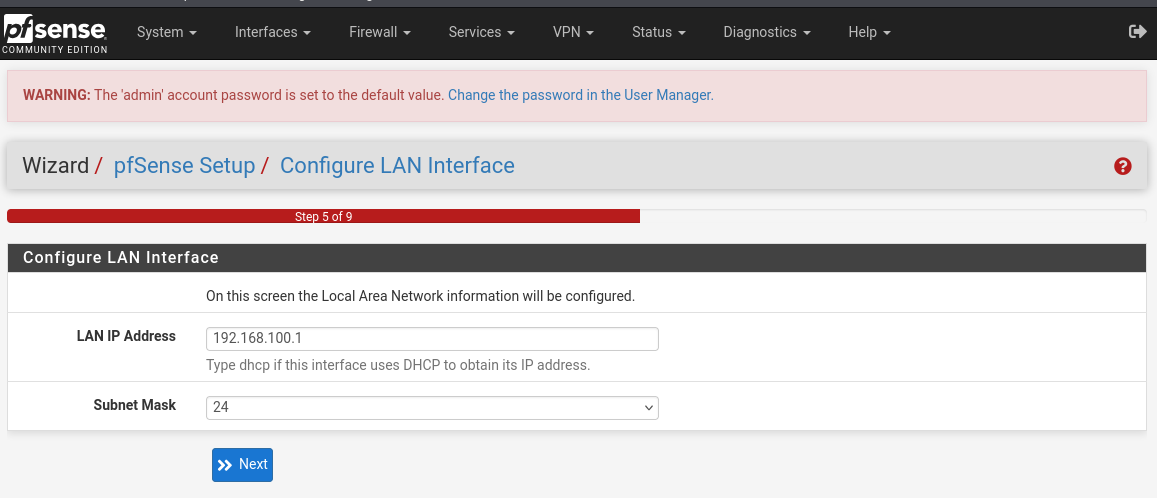
Las credenciales por defecto son “admin” para el usuario y contraseña.

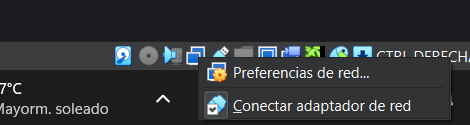


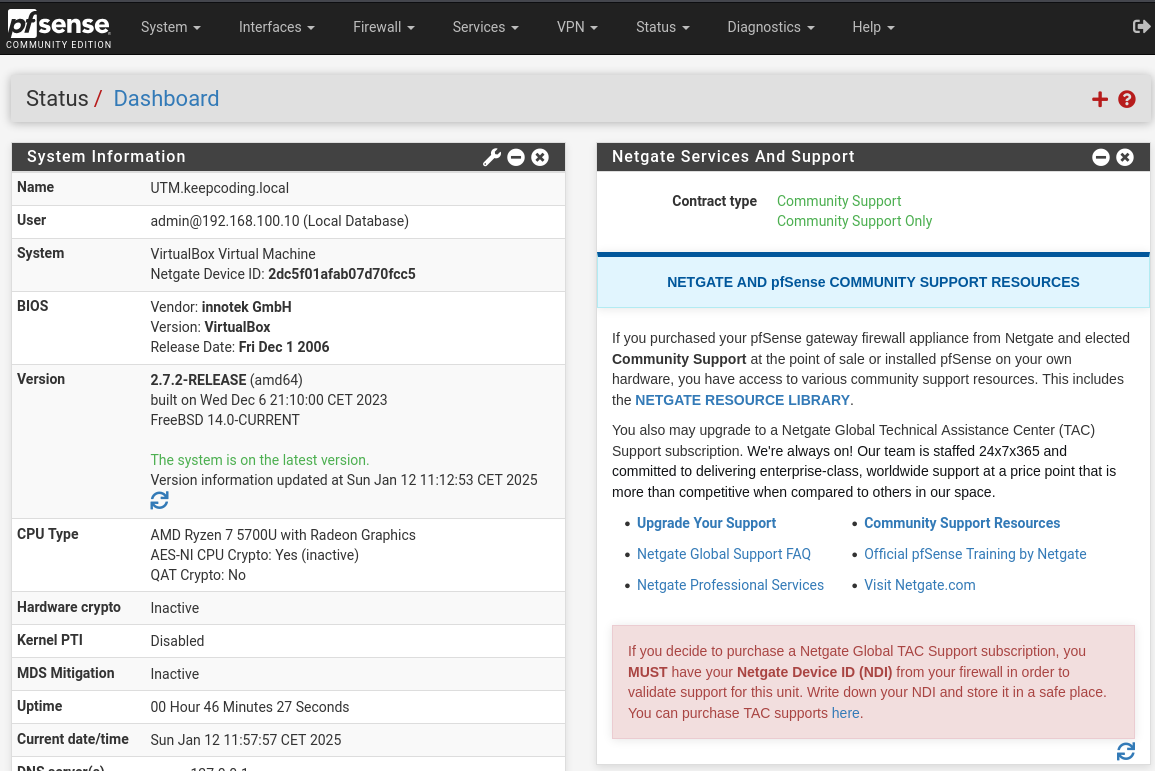


Debemos liberar los bloqueos de WAN para evitar conflictos con nuestra red interna, ya que trabajaremos solo con tráfico interno.

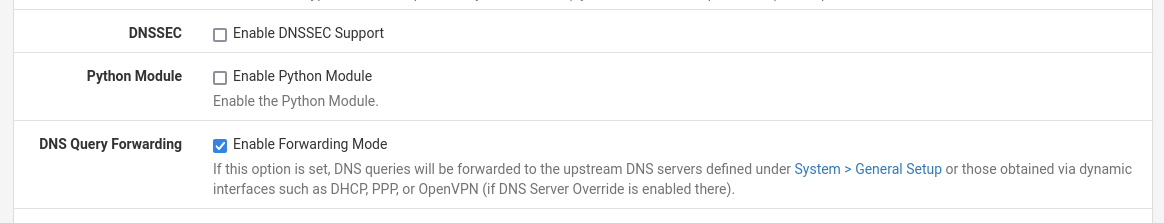


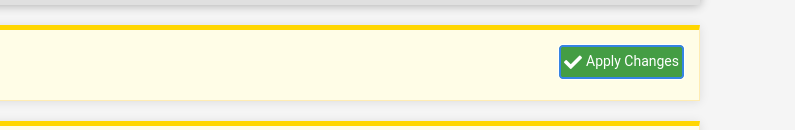
Cambiaremos la IP de LAN para evitar conflictos con la red de nuestro hogar.

Definimos una nueva clave de acceso, nuestras nuevas credenciales serán “user: admin | pass: 123456”. Refrescamos la página, desconectamos y conectamos nuevamente la red.

Accedemos con la nueva IP asignadas y las credenciales correspondientes. De esta manera logramos acceso al portal para iniciar el resto de las configuraciones.

Configuramos el servidor de DNS en nuestro pfSense. Para ello accedemos al apartado *ServicesDNS / ResolverGeneral / Settings*.

Quitamos el check del apartado “Enable DNSSEC Support” y colocamos el check en “Enable Forwarding Mode”.

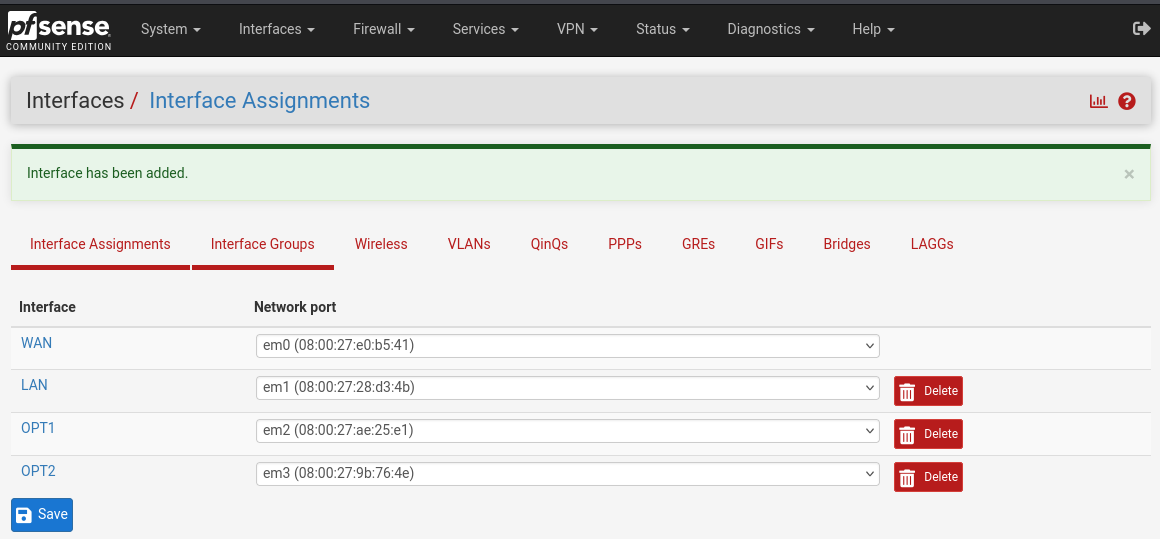
**IMPORTANTE “En todas las modificaciones que gestionemos, debemos aplicar los cambios desde el siguiente botón:”**

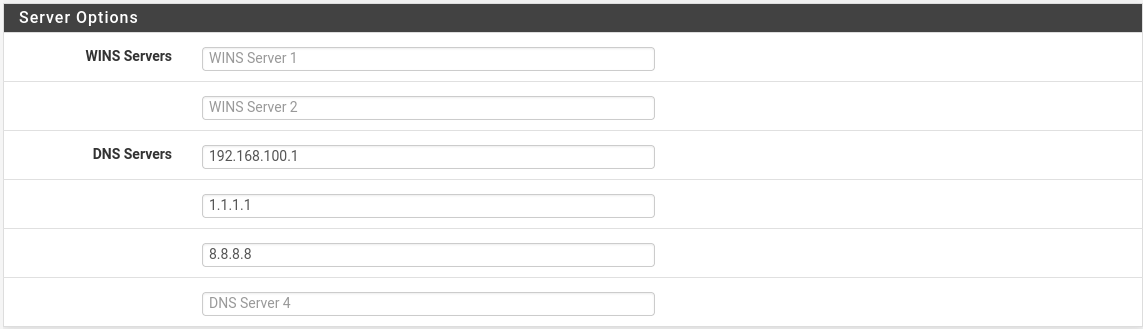
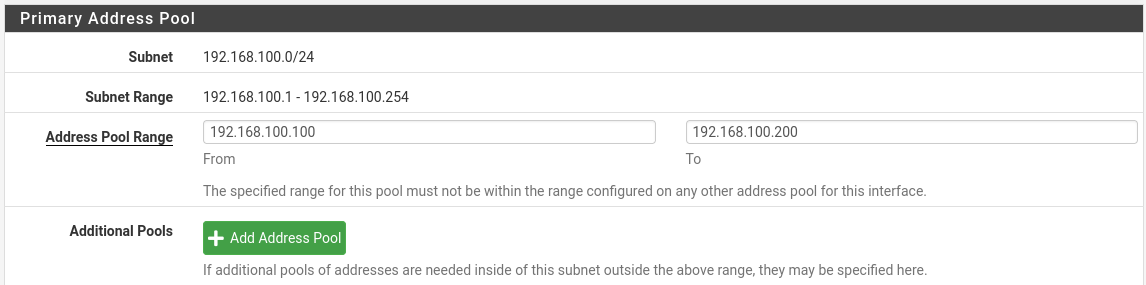
Configuraremos pfSense para que asigne IP dinámicos por cada Red generada. Usaremos datos conocidos para la configuración:

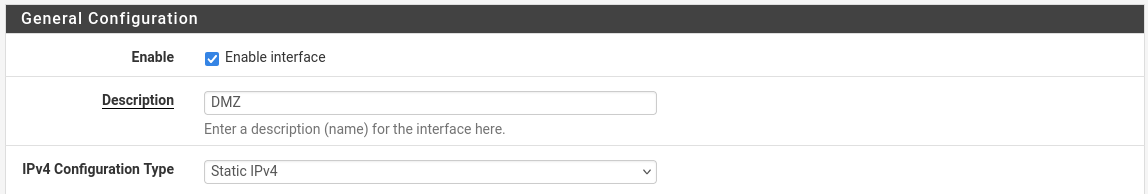
LAN: 192.168.100.1/24 Rango DHCP: 192.168.100.100 - 192.168.100.200

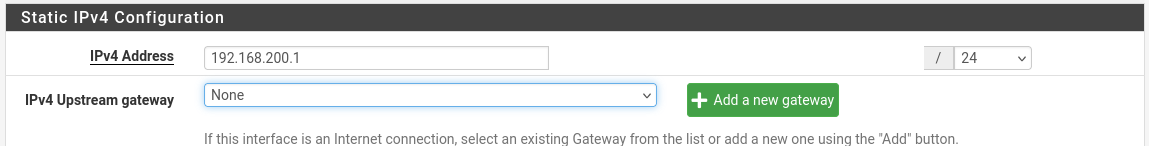
DMZ: 192.168.200.1/24 Rango DHCP: 192.168.200.100 - 192.168.200.150

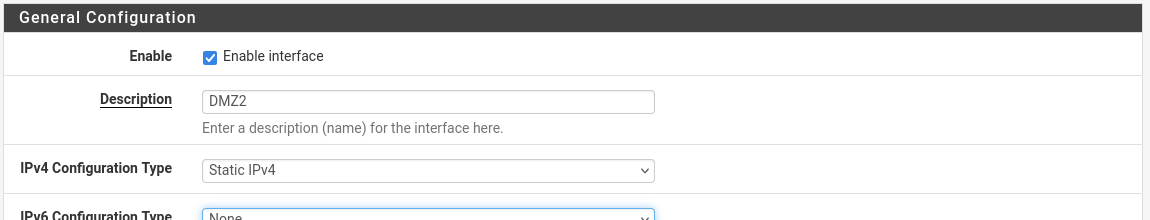
DMZ2: 192.168.250.1/24 Rango DHCP: 192.168.250.100 - 192.168.250.150

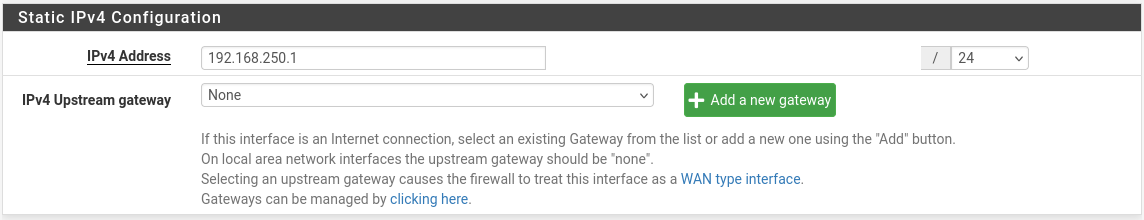
Habilitamos todas las interfaces a utilizar desde el apartado *InterfacesInterface / Assignments*.

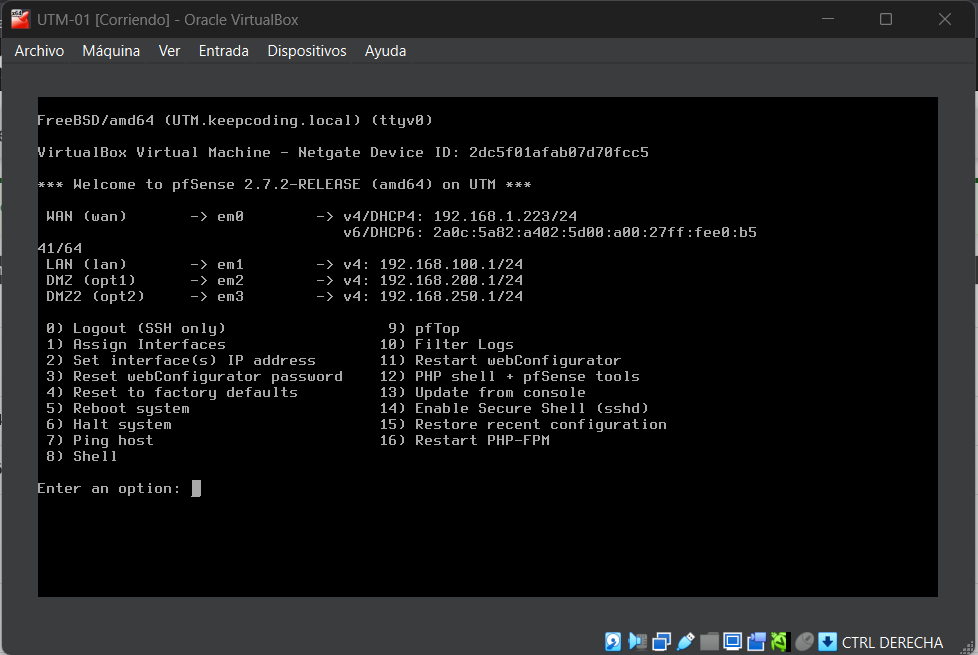
Desde el apartado *ServicesDHCP / ServerLAN* realizaremos la configuración de LAN con los parámetros antes mencionados:

Desde el apartado *Interfaces / OPT1 (em2)* realizaremos las configuraciones para la red DMZ



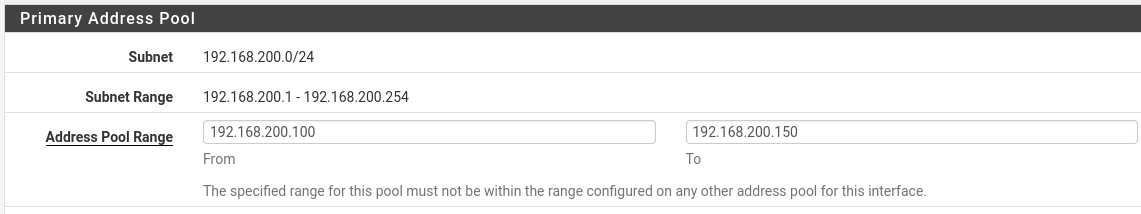
Aplicamos los cambios correspondientes en el apartado *Interfaces / OPT2 (em3)* para Red DMZ2

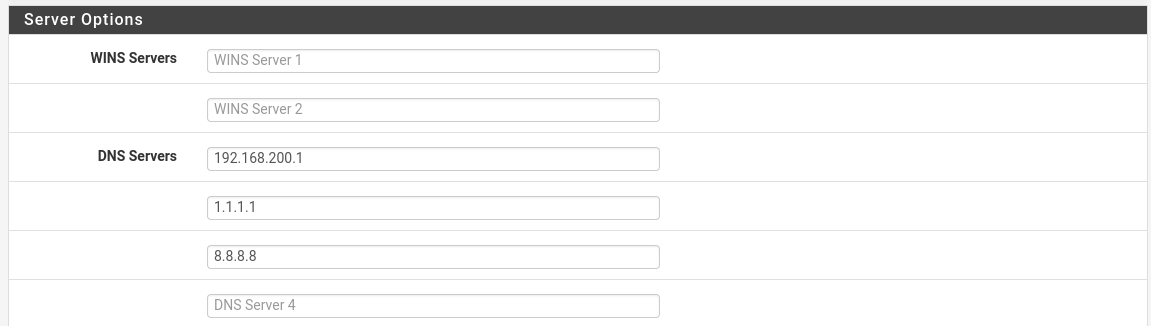


Verificamos que hemos configurado las redes de forma correcta desde nuestra maquina UTM-01

Validando esta configuración, aplicamos los parámetros restantes desde el apartado *Services / DHCP Server* para las redes DMZ y DMZ2.

Parámetros para DMZ

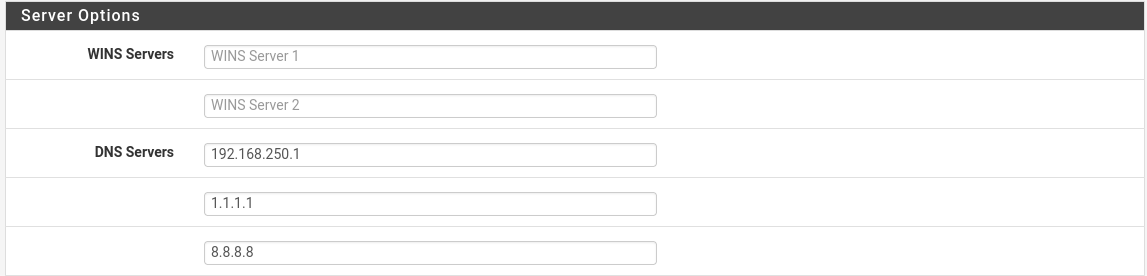


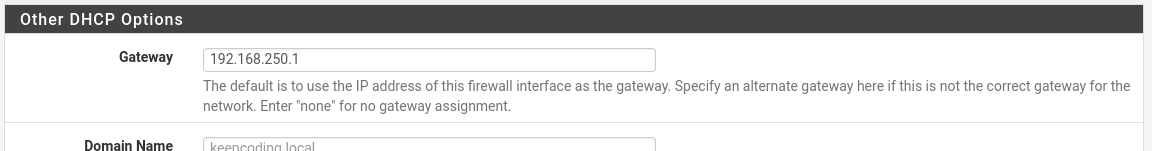




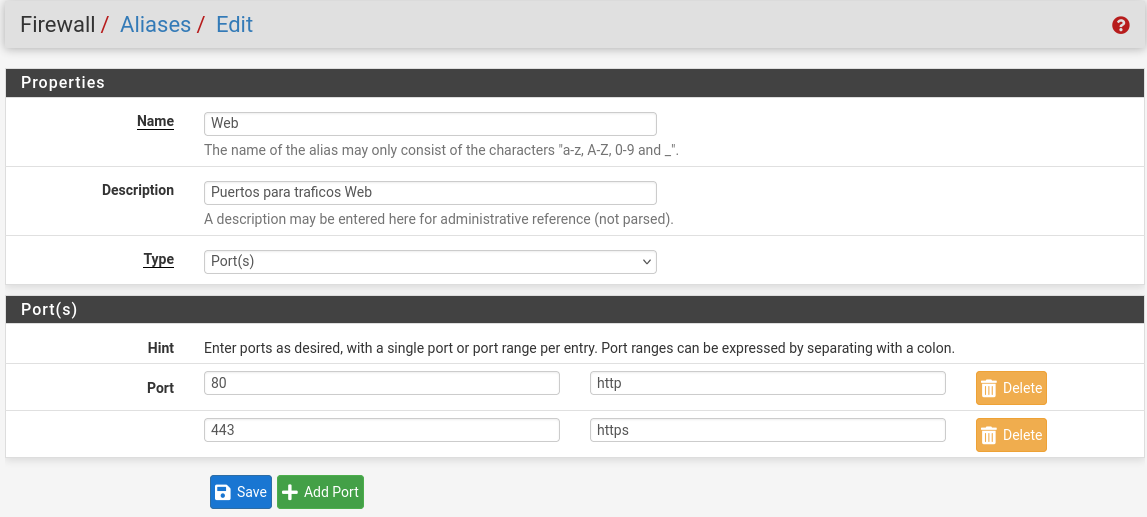
Parámetros para DMZ2





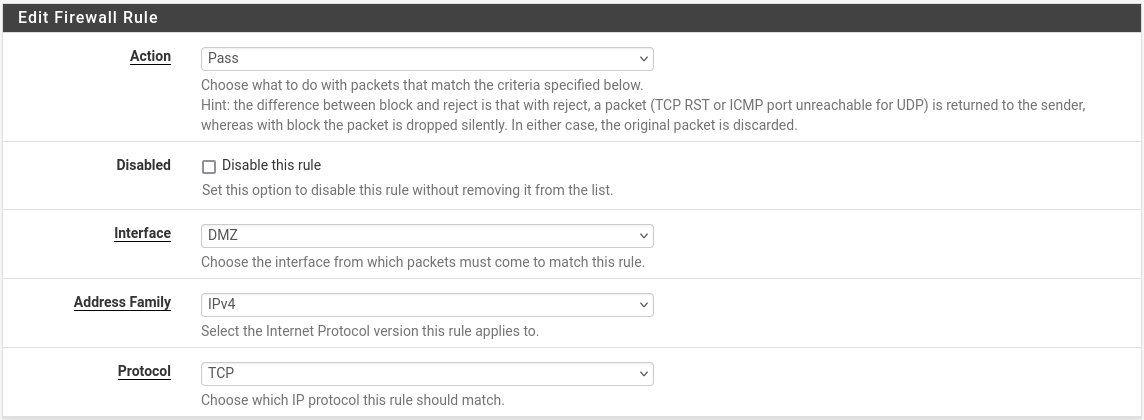


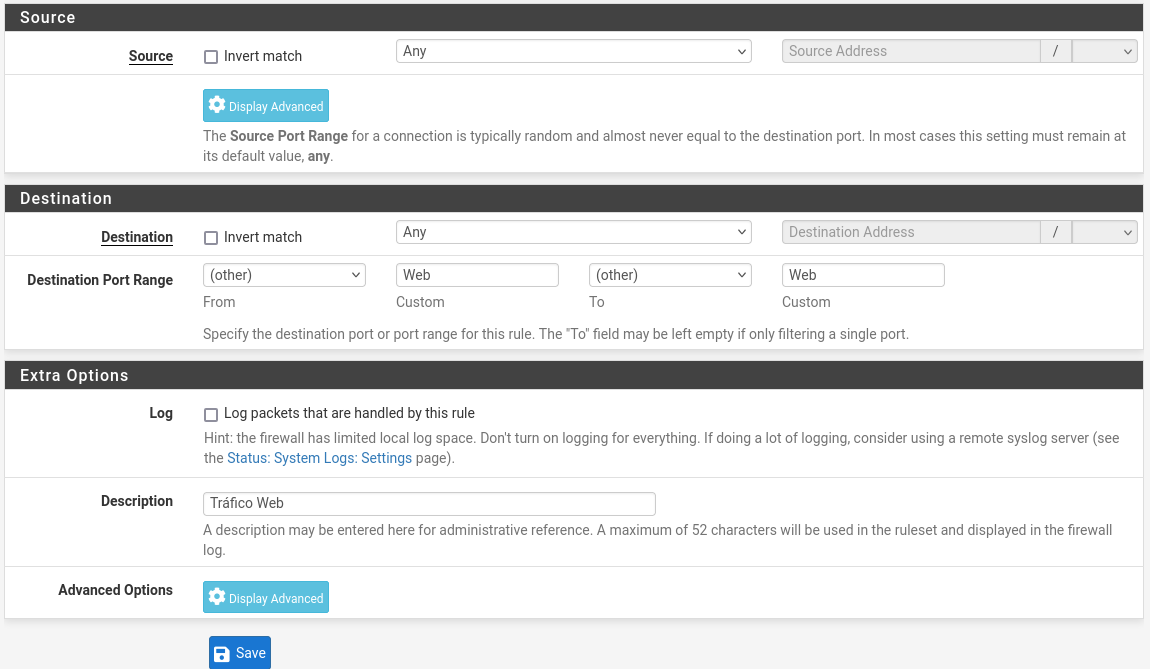
**Definimos las normas de Firewall para cada Red creada.**

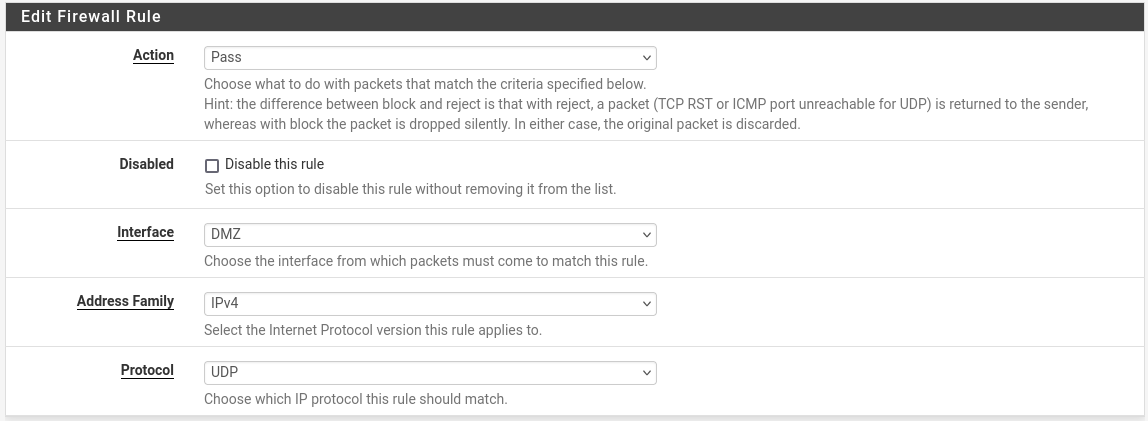
Previamente a esta configuración, asignaremos un alias para simplificar los parámetros en las normas de firewall. Desde el apartado *Firewall / Aliases / Ports* agregamos el alias “Web” con los puertos 443 y 80 para configurar los accesos a internet.

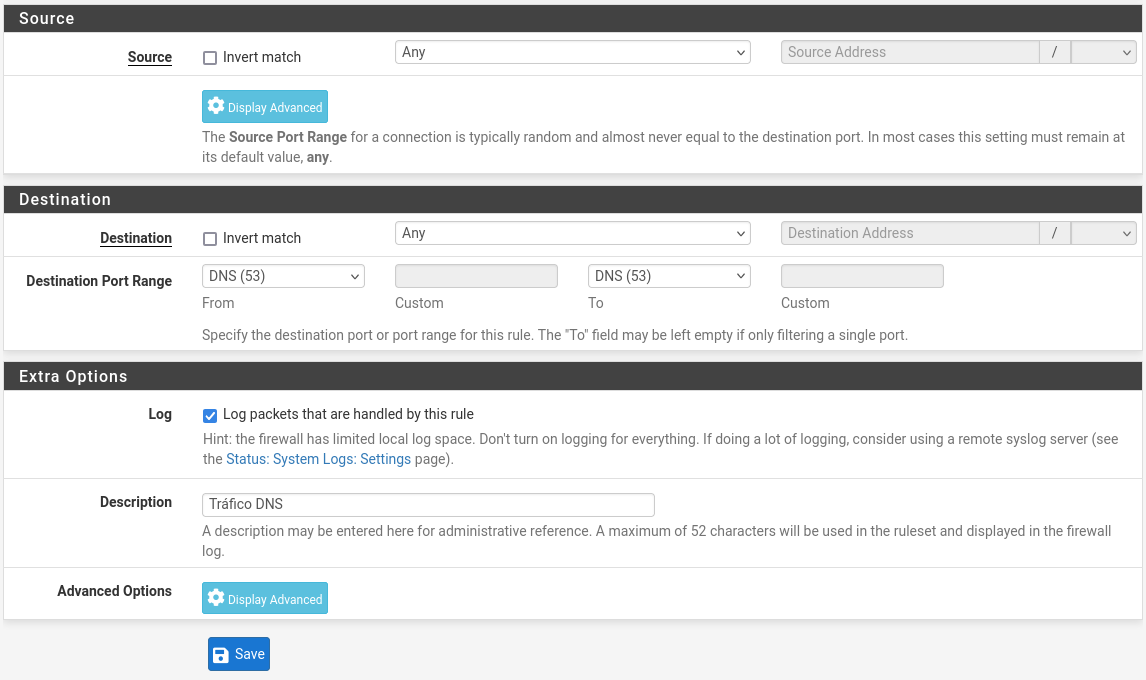
Utilizaremos las configuraciones por default de la Red LAN, por lo que aplicaremos en principio, las configuraciones básicas para las Redes DMZ y DMZ2.

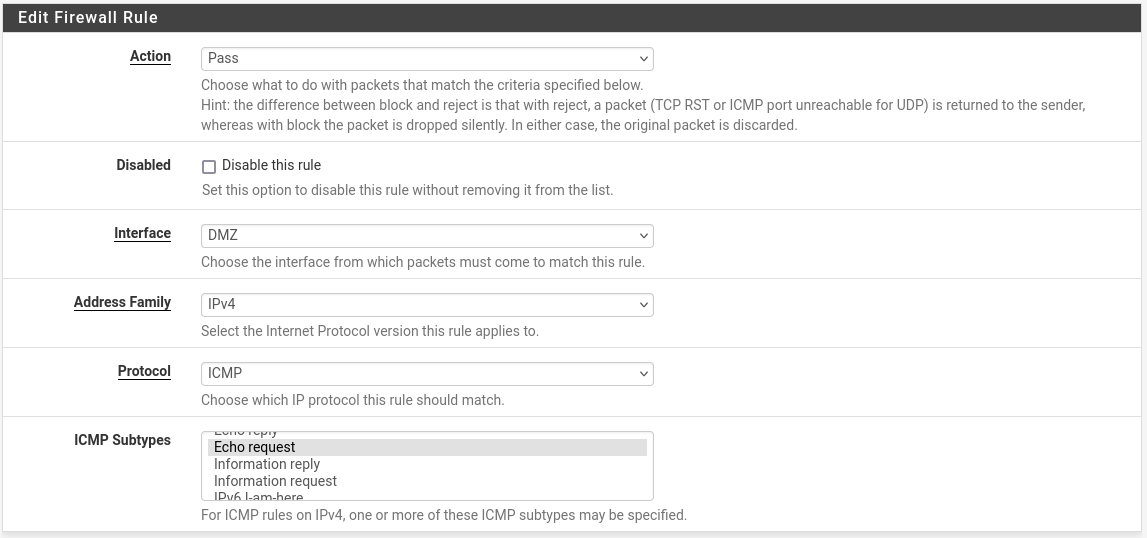
Desde el apartado *FirewallRulesDMZ* asignamos los parámetros en la Red DMZ.

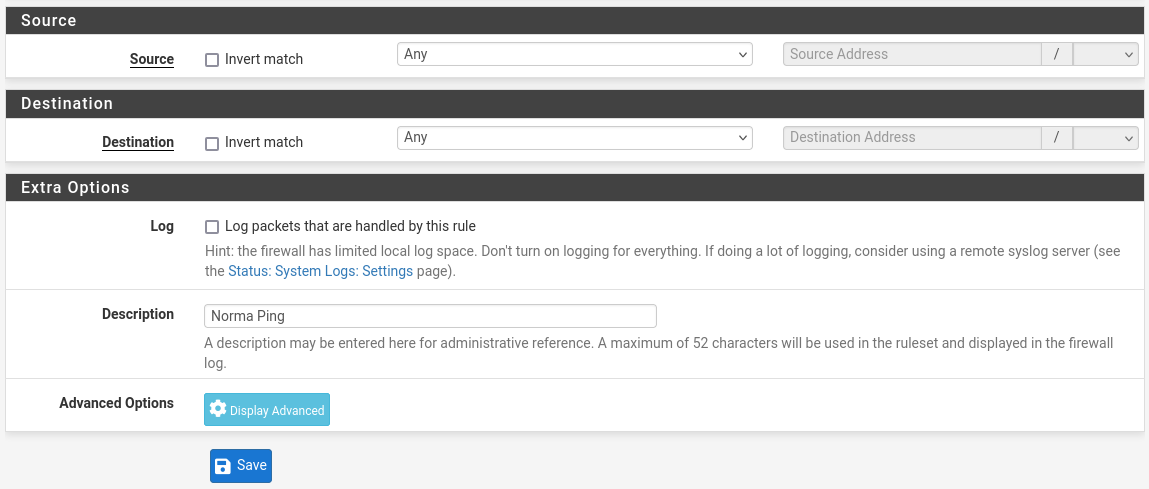
Aplicamos las normas para el acceso a internet.

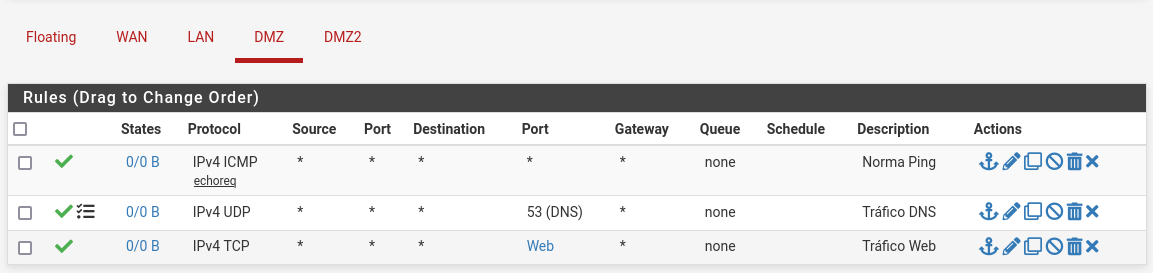


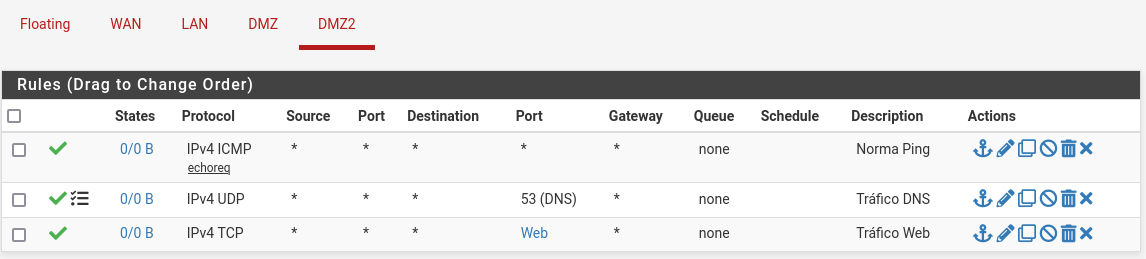
Aplicamos las normas para las configuraciones de DNS.

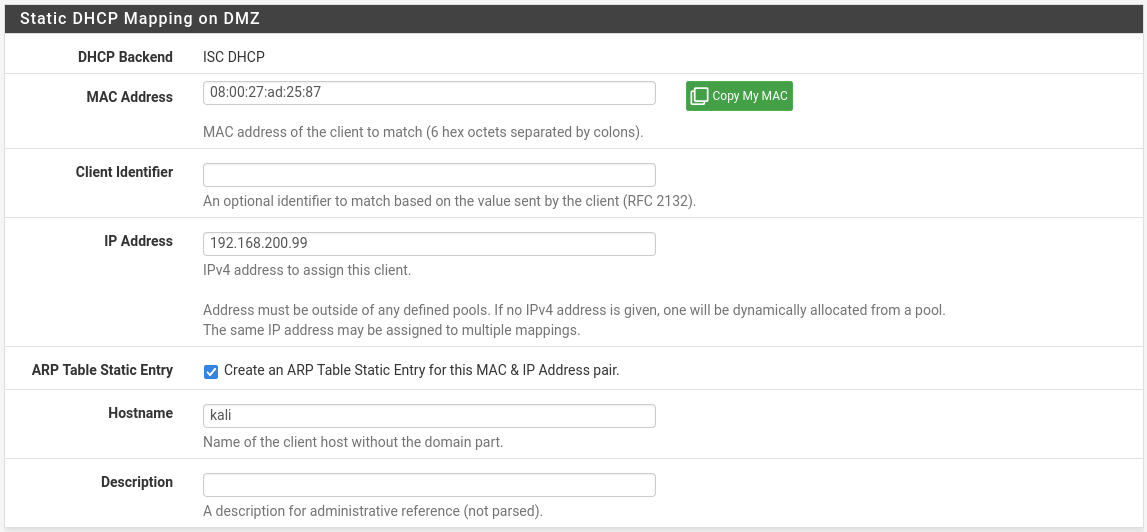
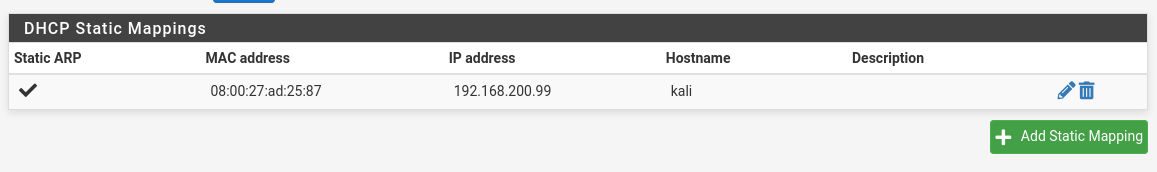


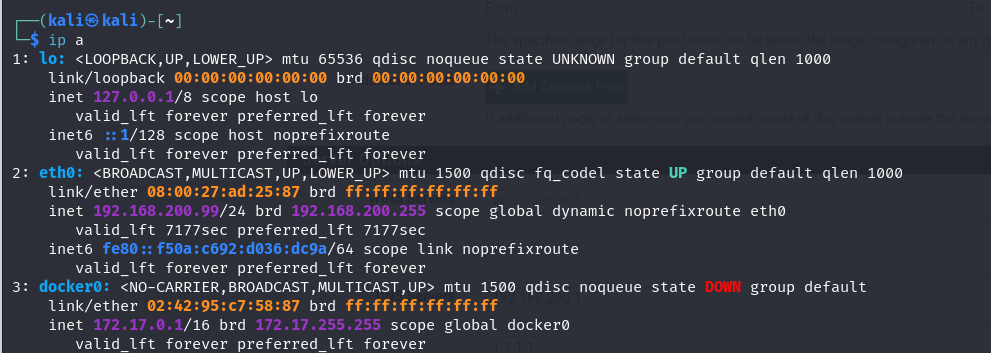
Aplicamos las normas para las configuraciones de PING. 



Las reglas definidas para la Red DMZ son las siguientes: 

Replicamos las mismas configuraciones sobre las Red DMZ2 antes de comenzar a gestionar las restricciones sobre la Red DMZ quedando de las siguiente manera: 

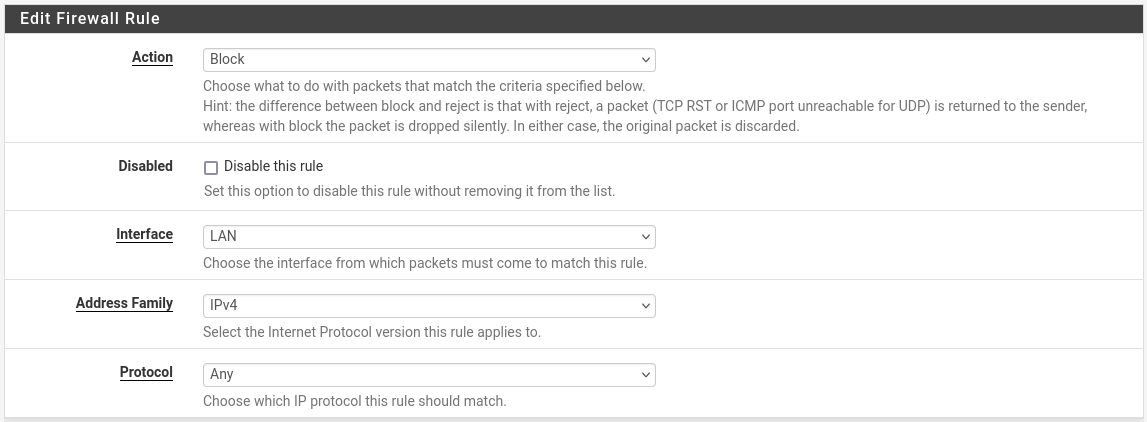
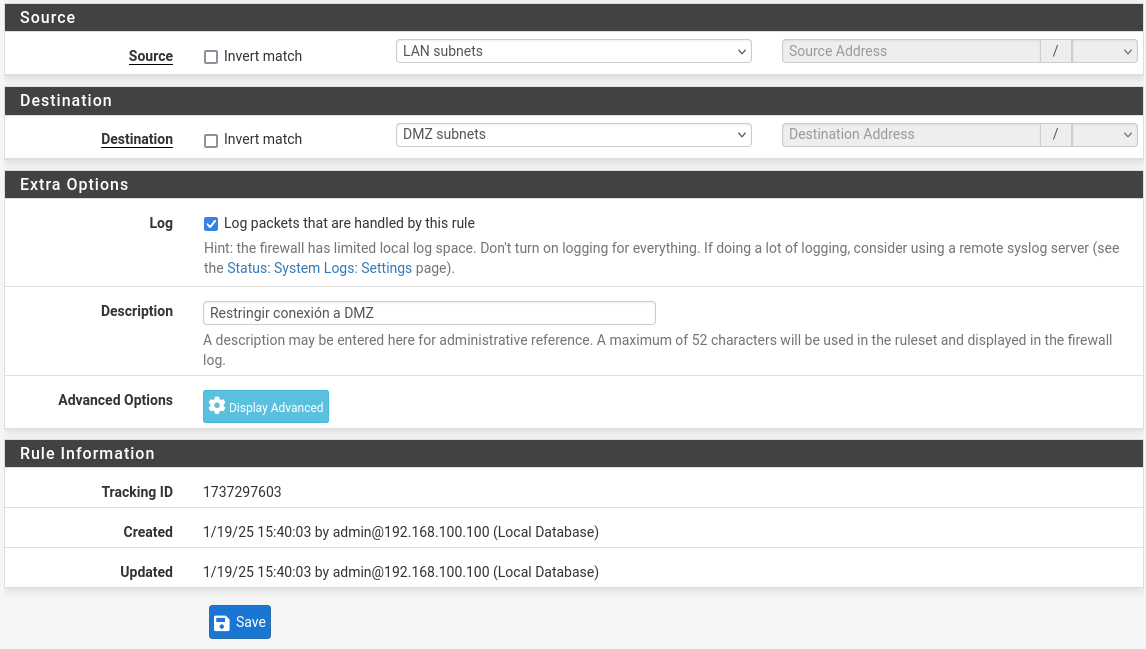
Configuramos una IP estática sobre la Red DMZ para hace accesible desde el exterior el Honeypot. Asignaremos el IP “192.168.200.99” desde el apartado *Services / DHCP Server / DMZ / Static Mapping / Edit*. 

Evidenciamos la configuración desde Kali conectado a la Red DMZ en el terminal. 

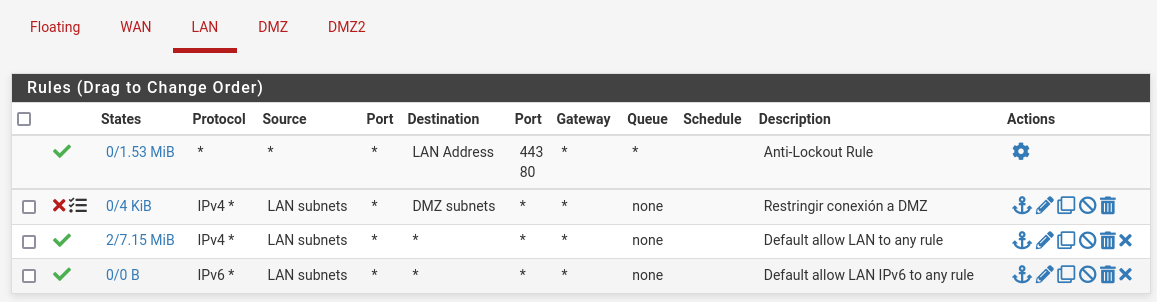
**Restricciones de Red.**

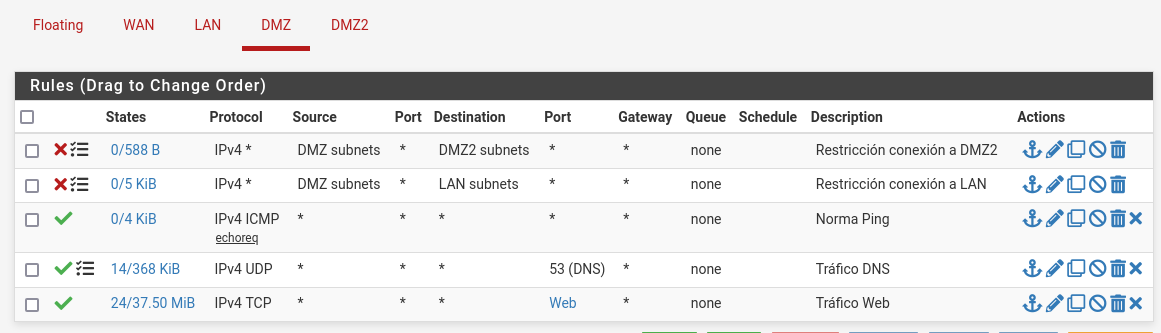
Configuramos nuestro Firewall desde pfSense para que DMZ no tenga acceso al resto de las redes.

Utilizaremos los siguientes parámetros para configurar cada Red.

**** ****

Quedando configurados de la siguiente manera:







LAN no podrá conectarse con DMZ, pero si con WAN y DMZ2

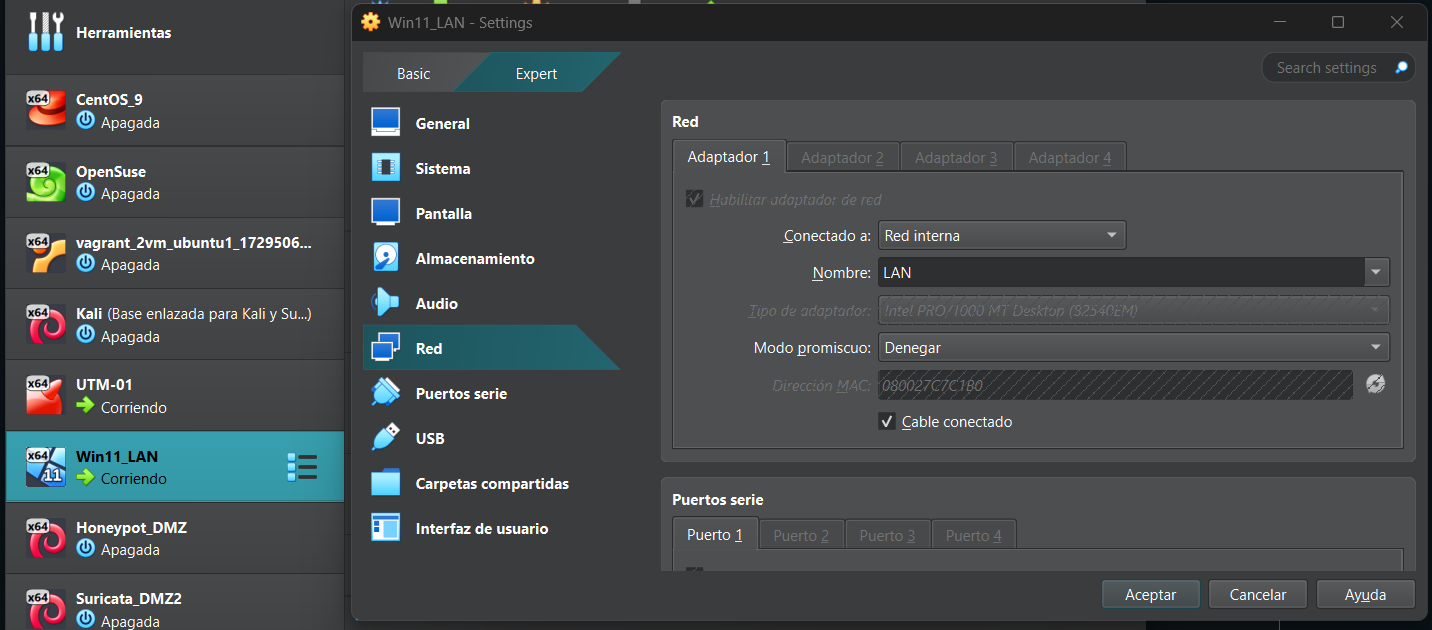
DMZ2 no podrá conectarse con DMZ, pero si con LAN y WAN

DMZ no podrá conectarse con LAN y WAN

WAN podrá conectarse con LAn, DMZ y DMZ2

**2. En la red LAN debe haber un equipo Windows 11 que envíe logs al servidor de Elastic.**

Para generar la conexión dentro de nuestro UTM-01, instalaremos una ISO de Wind11 en nuestra Virtual Box, dejando las siguientes configuraciones.



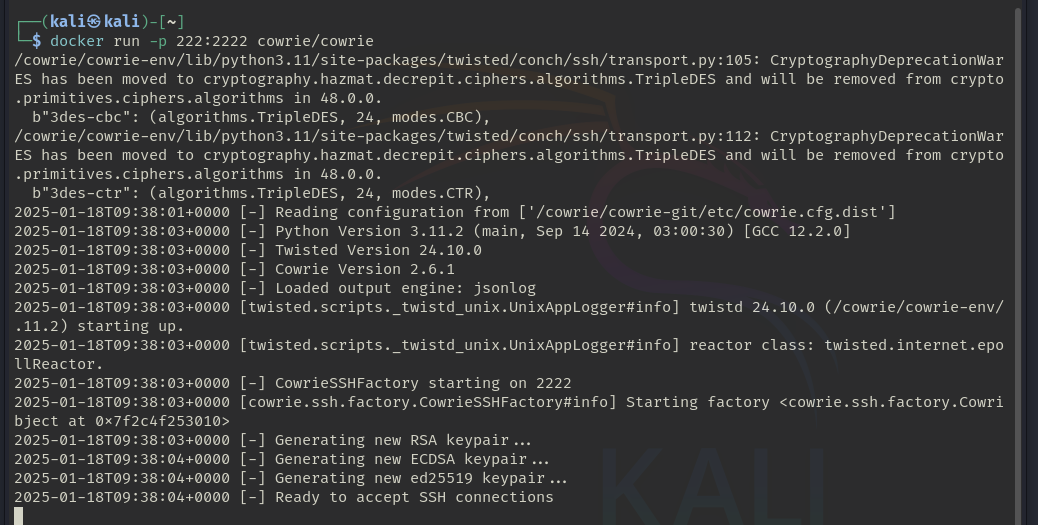
**3. En la red DMZ debe haber un honeypot que envíe los logs al servidor de Elastic.**

**1. Este honeypot no debe tener acceso a ninguna red interna (LAN, DMZ2...) y debe ser**

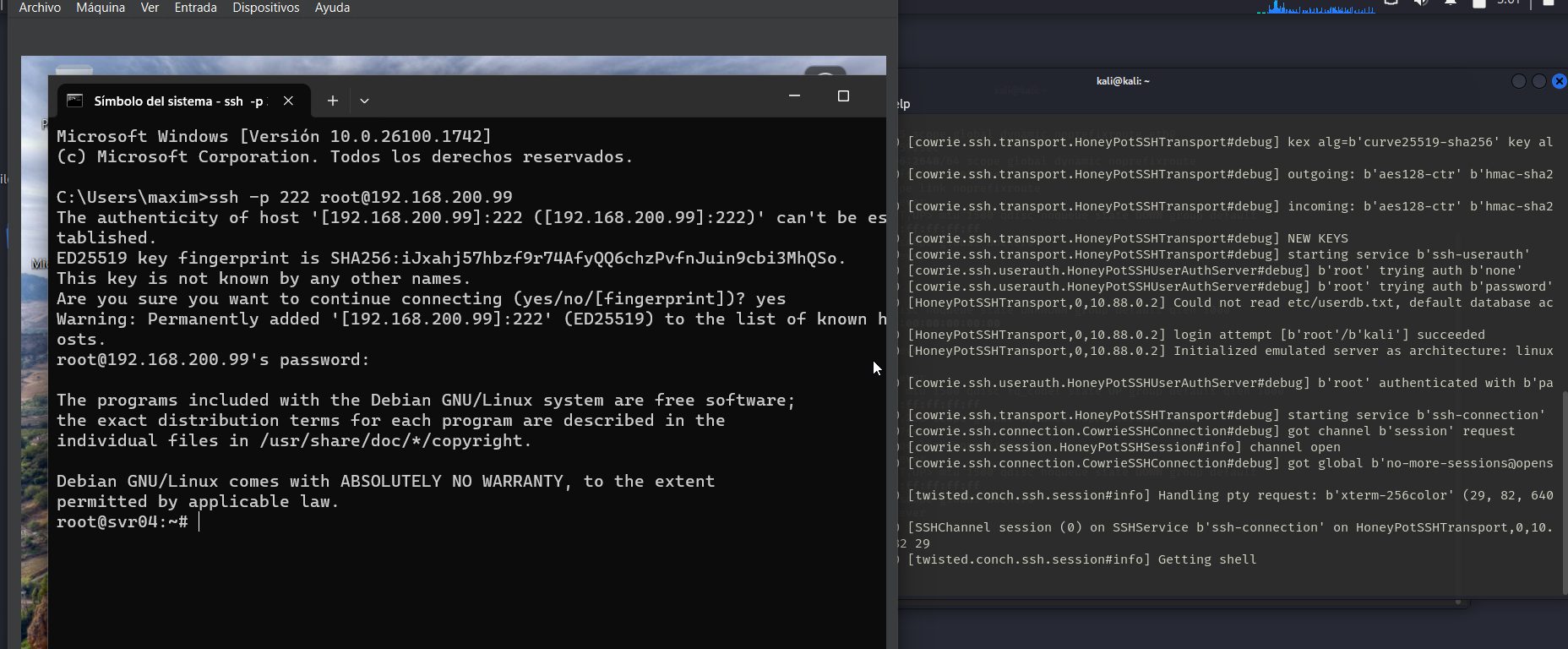
**accesible desde el exterior (red WAN) en ambos sentidos.**

En este punto utilizaremos Cowrie como Honeypot por su facilidad de configuración. Desde los repositorios Docker con el siguiente comando:

*docker run -p 222:2222 cowrie/cowrie*



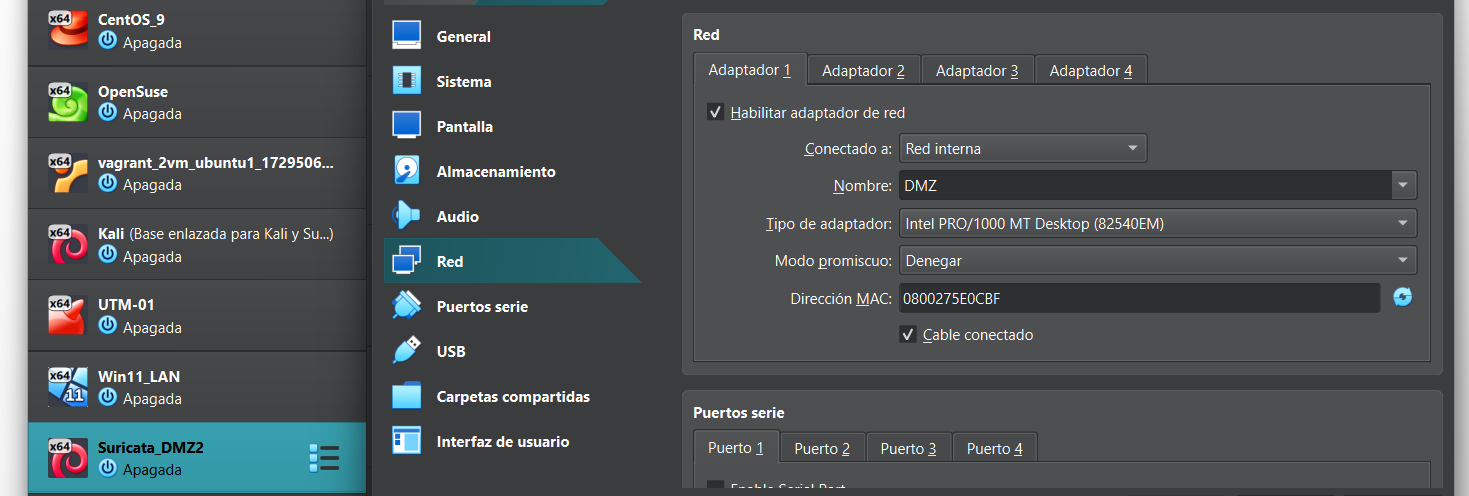
Para validar el funcionamiento correcto, conectaremos nuestro Wind11 con el protocolo SSH usando el siguiente comando *ssh -p 222 root@192.168.200.99* (Se asignó como IP estático previamente)



**4. En la red DMZ2 debe haber otra fuente diferente de logs a las dos mencionadas anteriormente.**

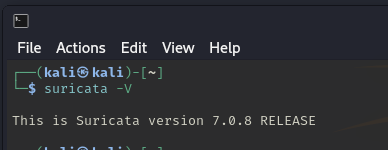
**Se propone Suricata o Apache Server como posibles fuentes, pero se deja a elección del alumno.**

Creamos nuestra máquina virtual denominada “Suricata\_DMZ2”, para generar nuestro log desde Suricata. Configuramos tipo de Red Interna / DMZ2.



Dentro de la máquina actualizaremos los paquetes e instalaremos Suricata con los siguientes comandos:

*sudo apt update && sudo apt install suricata*



Antes de la lanzar la herramienta vamos a generar las reglas de monitoreo/log. Vamos a seguir el orden de los siguientes comandos en el terminal de la máquina virtual.

*sudo -s* para gestionar con permisos de Root.

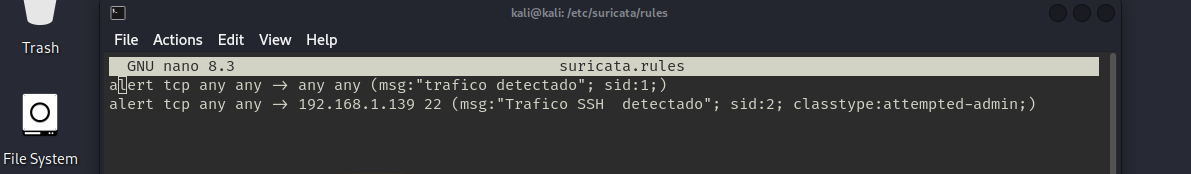
*cd /etc/suricata/rules* para movernos al directorio de Reglas.

*touch suricata.rules* para generar el archivo donde definiremos los parámetros.

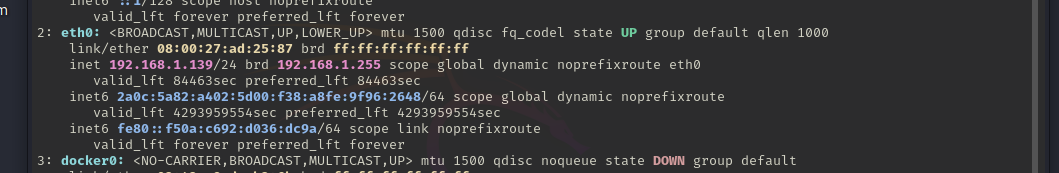
Asignaremos las siguientes normas:

*alert tcp any any -> any any (msg:"trafico detectado"; sid:1;)*

*alert tcp any any -> 192.168.1.65 22 (msg:"Trafico SSH detectado"; sid:2; classtype:attempted-admin;)*



Considerar que la segunda regla de identificara el IP de nuestra máquina virtual, lo consultamos en un nuevo terminal con el comando *ip a*.

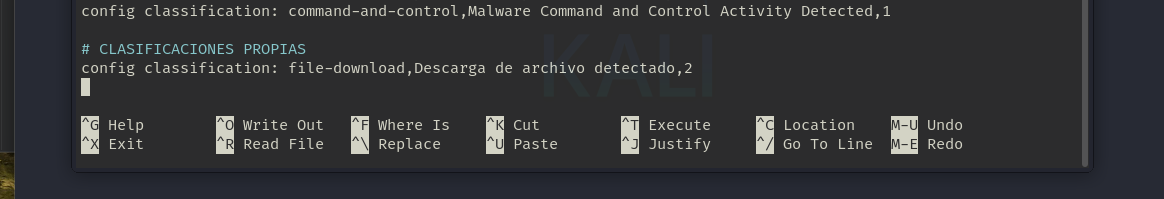


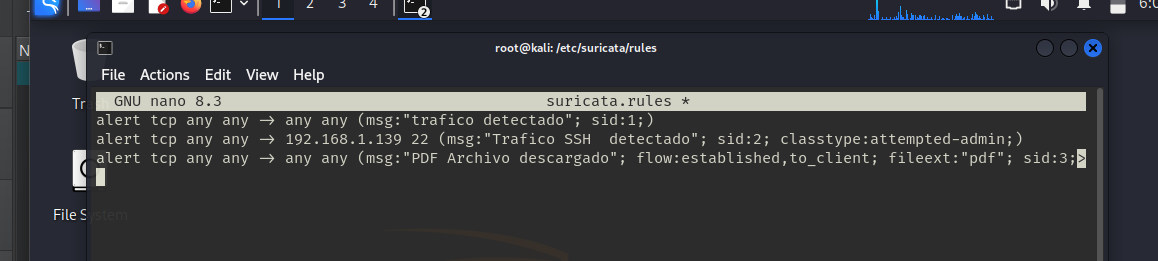
Nos movemos al directorio */etc/suricata* para configurar el archivo *suricata.yaml* de la siguiente manera:



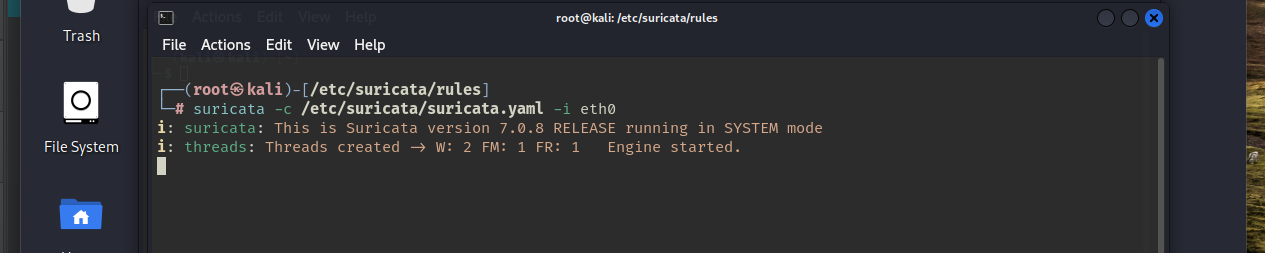
Asignamos la ruta *default-rule-path: /etc/suricata/rules* y comentamos la siguiente fila *#default-rule-path: /var/lib/suricata/rules*. Tendremos este acceso como backup ante conflictos futuros.

Agregaremos nuestras clasificaciones propias desde la ruta */etc/suricata* y asignamos una nueva regla en el archivo *suricata.rules*:





Iniciamos Suricata desde con el comando *suricata -c /etc/suricata/suricata.yaml -i eth0*

Si las configuraciones son correctas, la terminal mostrará el siguiente mensaje.

**5. El servidor de Elastic debe recibir, almacenar y poder visualizar los logs del honeypot, el**

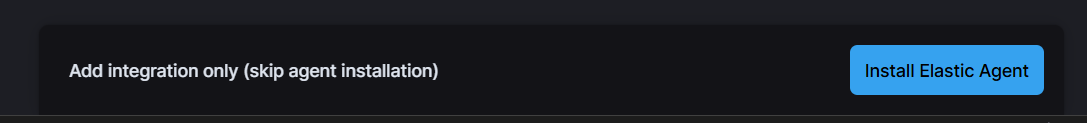
**Windows 11 y la fuente elegida ubicada en la DMZ2.**

Realizamos las configuraciones de Elastic para enlazar cada máquina virtual y gestionar los logs de cada una.

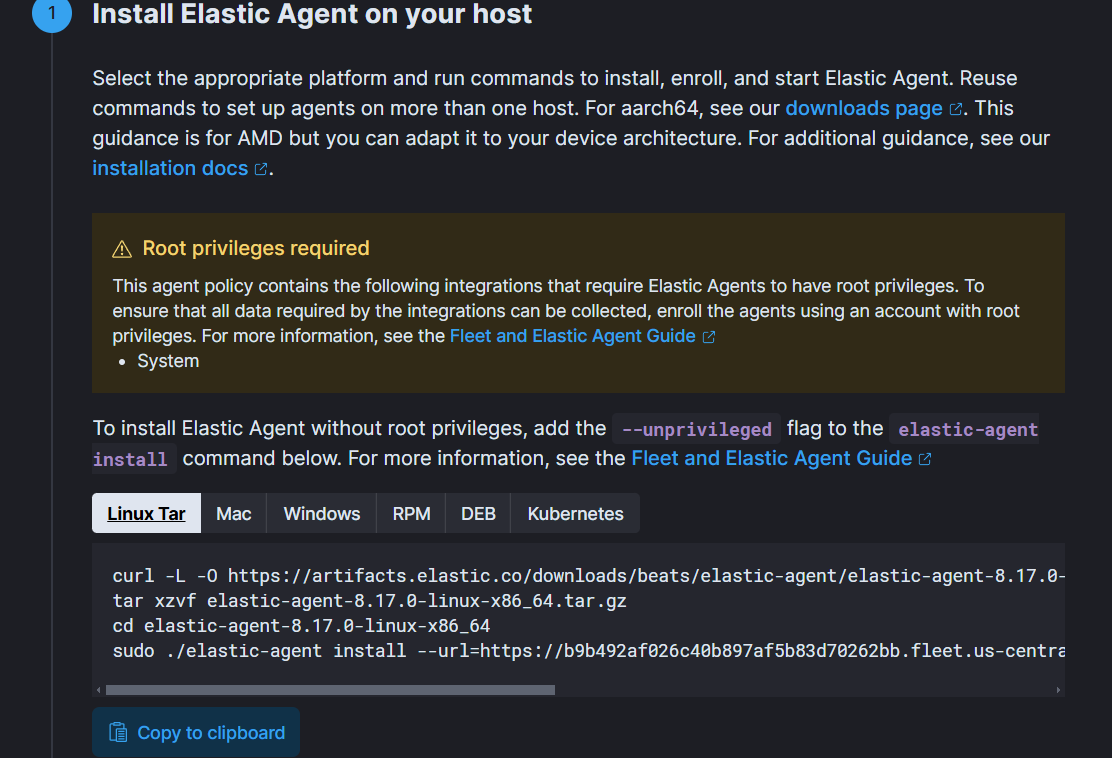
Desde <https://www.elastic.co/es/cloud> creamos nuestra cuenta con los datos solicitas, utilizaremos un usuario temporal para la práctica.

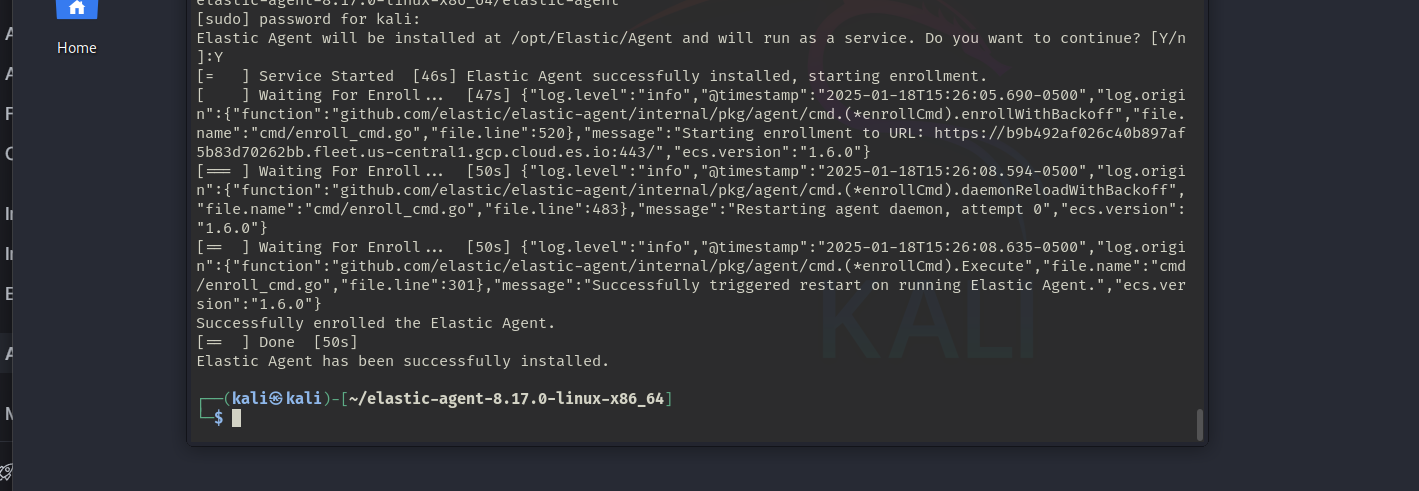
Una vez creada la cuenta comenzamos a configurar desde el apartado Assets > Agent policies > Create agente policy. *Evidenciamos las configuraciones para “Suricata/Linux” y replicaremos las configuraciones básicas para el resto de lokalis Agent.*

Para generar la integración utilizamos el apartado Assets > Agent policies > Add integration > Add Suricata. Pinchamos el botón emergente para acceder al enrolamiento.

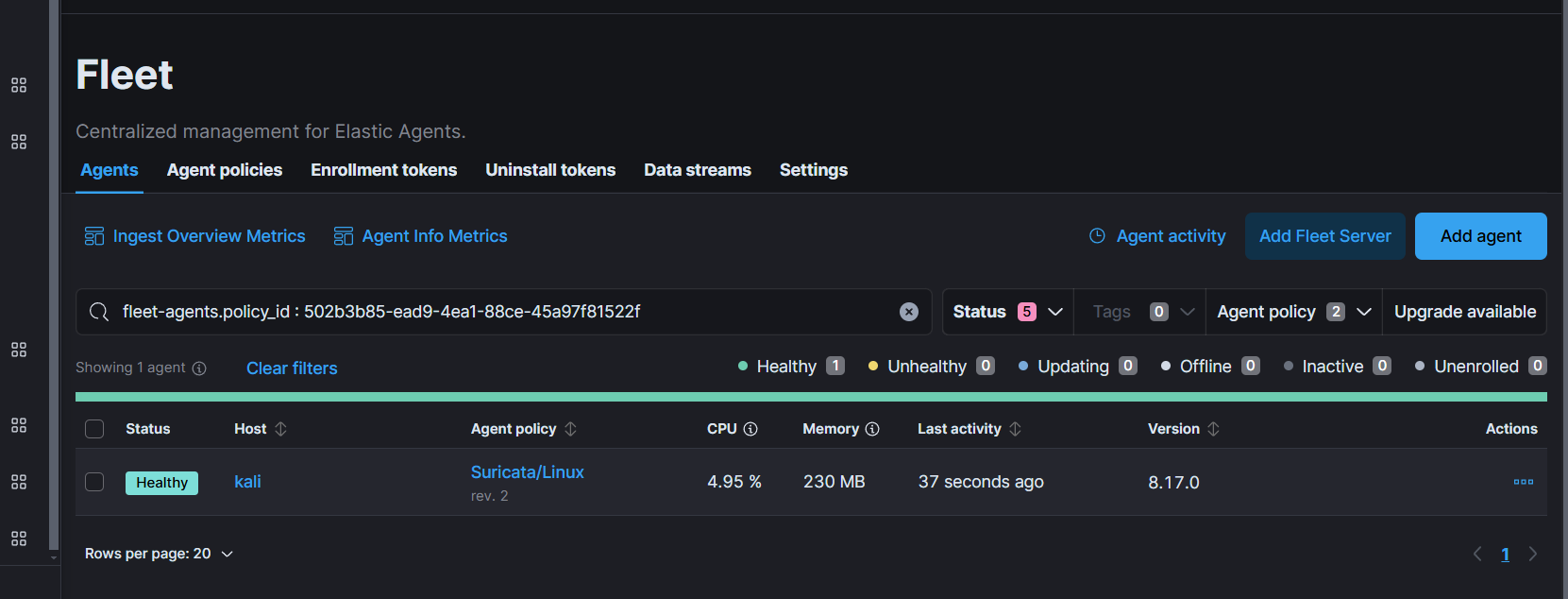


Seguimos las instrucciones para Linux Tar

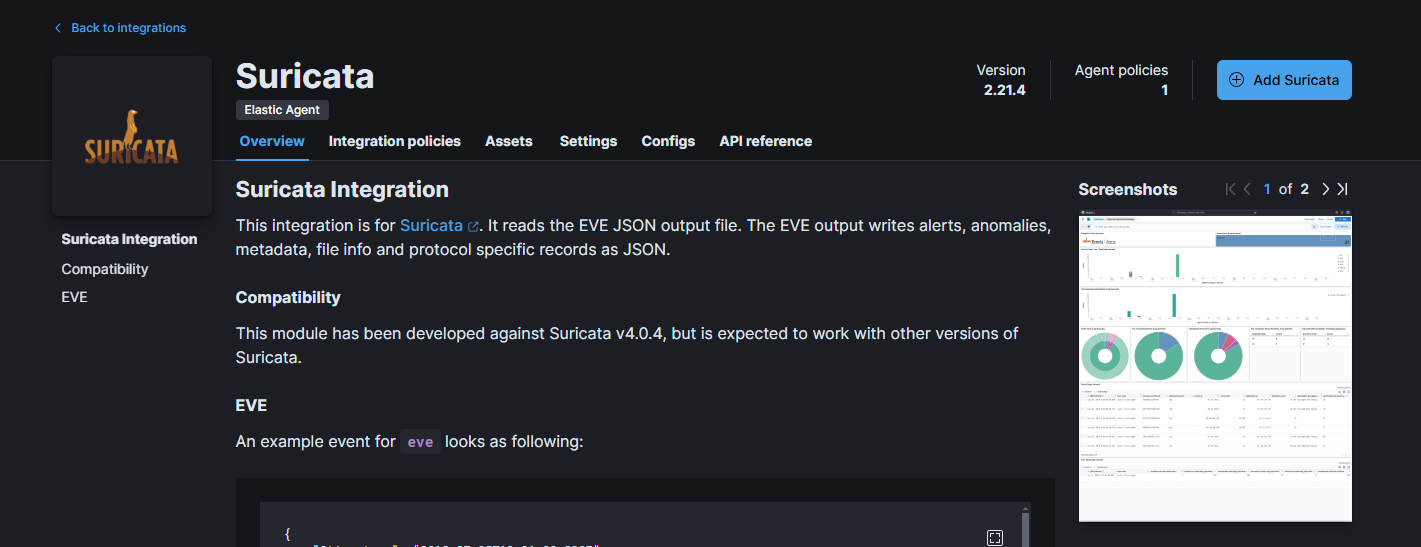


Desde nuestra terminal Suricata lanzamos el comando. 

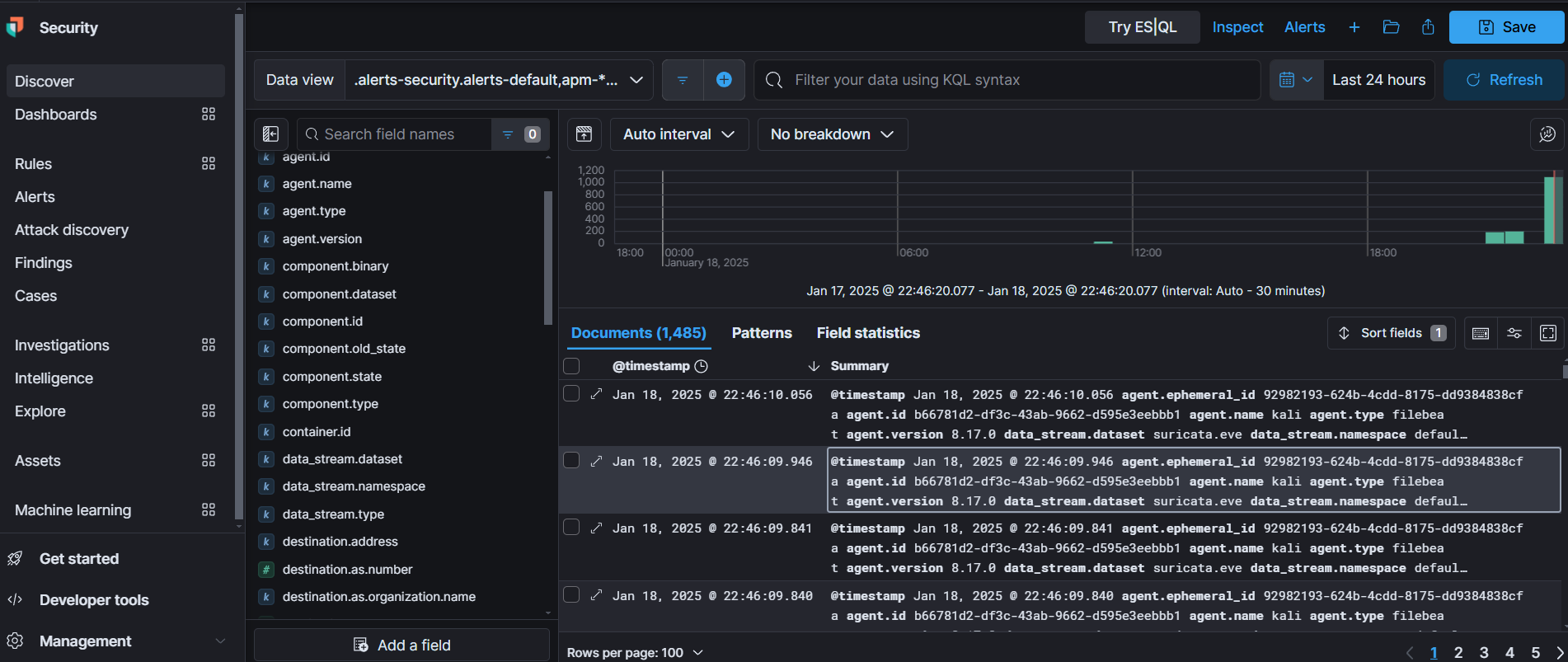
Una vez realizada la configuración encontraremos el siguiente panel.



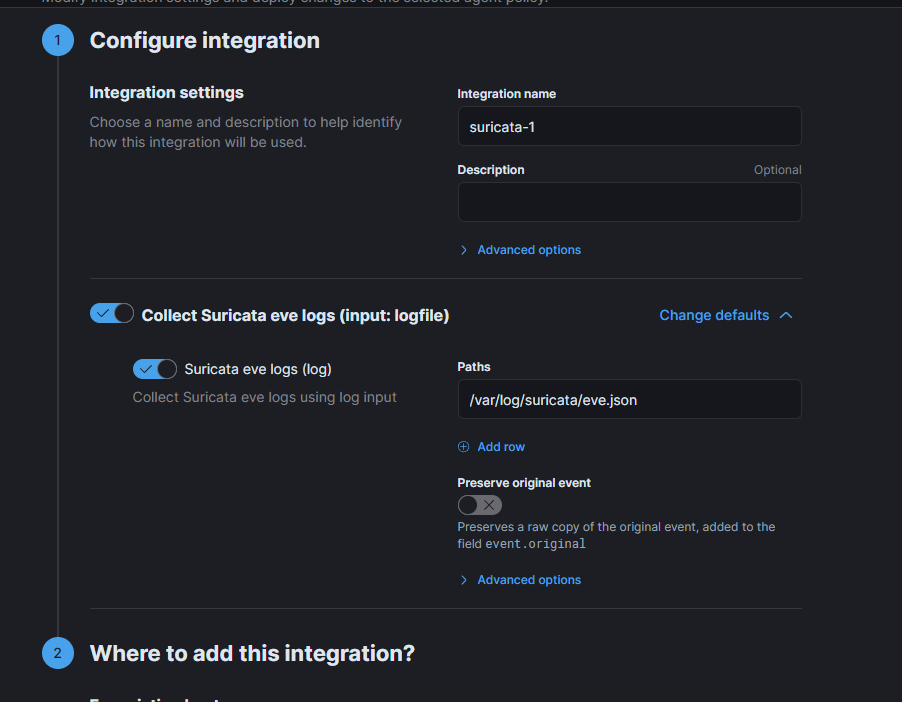
**Integraciones Suricata**

Para el Agent Suricata/Linux utilizaremos la siguiente integración. 

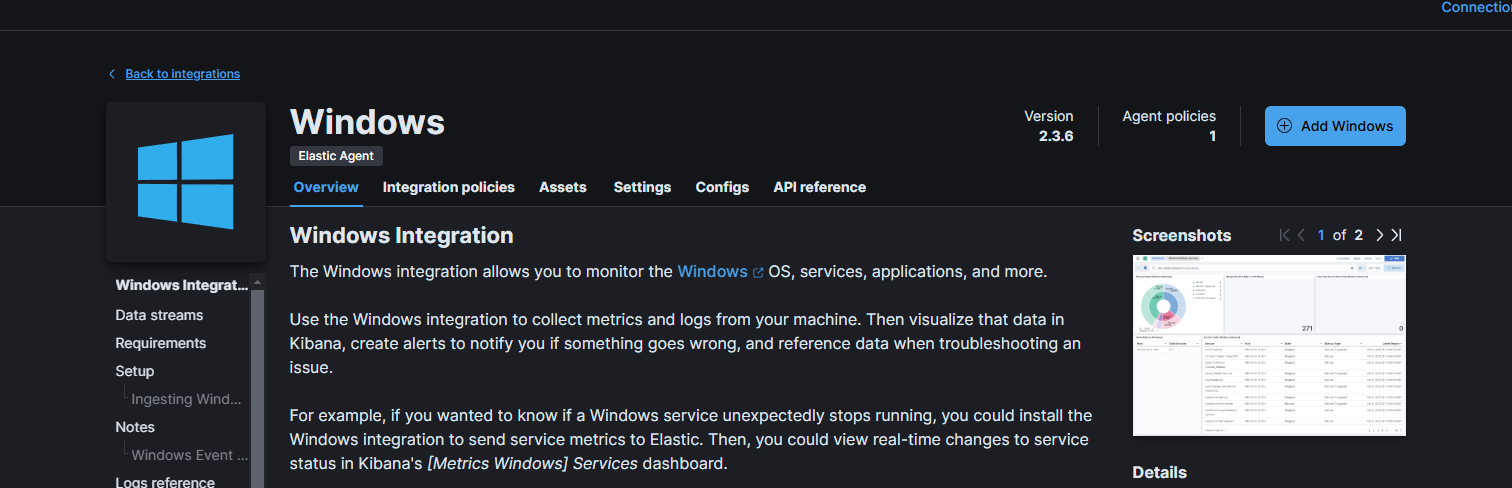
Desde el apartado Discover podremos ver los Log de nuestro Suricata enlazado en Elastic

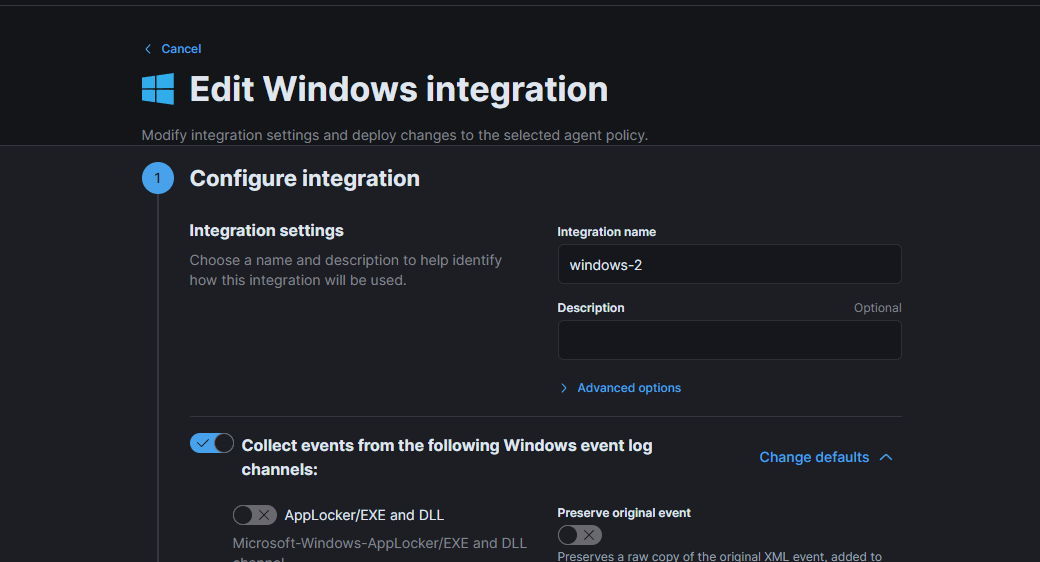


La integración ya dispone del path de logs, por lo que no tendremos que realizar modificaciones.



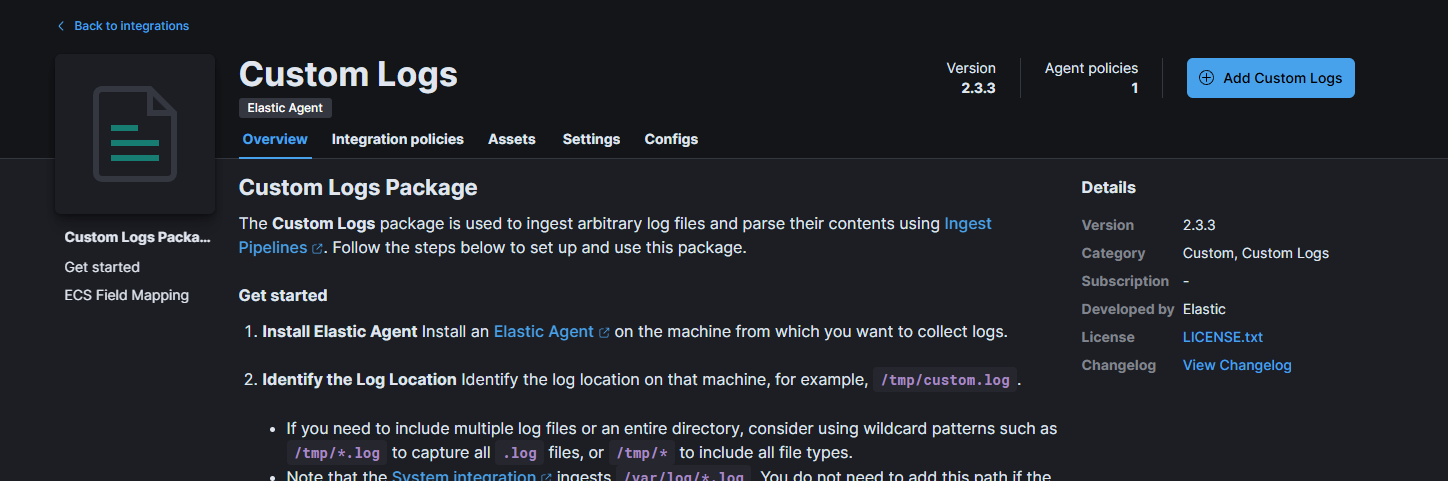
**Integraciones para Wind11**

Para el Agent Wind11 utilizaremos la siguiente integración. 

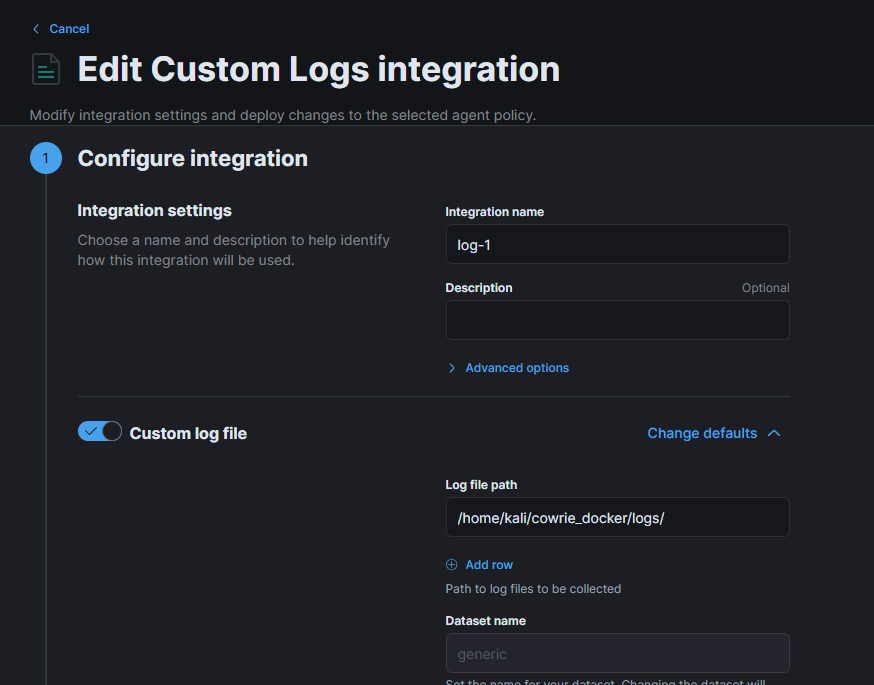
Aquí no será necesario modificar datos, ya que la configuración default es correcta.

**Integraciones para Honeypot**

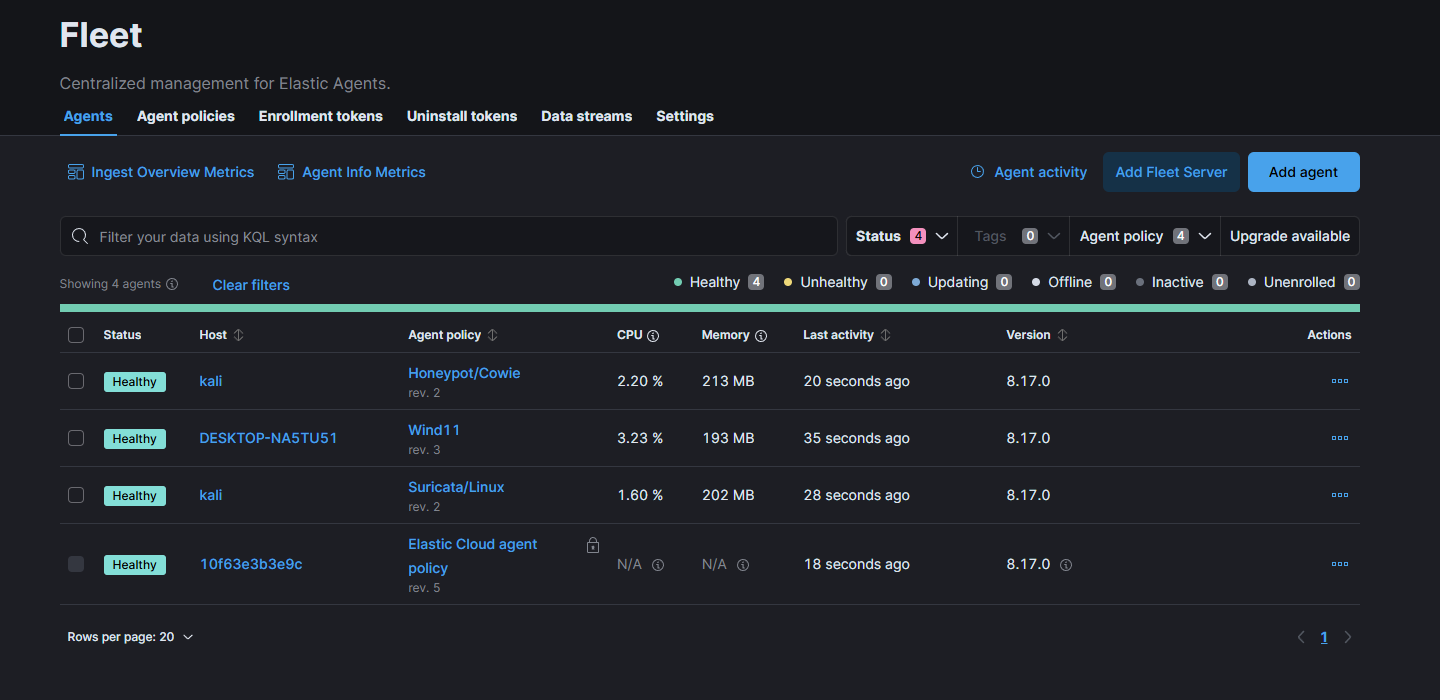
Para el Agent Honeypot utilizaremos la siguiente integración (custom)



Asignaremos el siguiente path con las configuraciones adicionales “/home/kali/cowrie\_docker/logs/”



Nuestra configuración queda evidenciada en el siguiente panel.



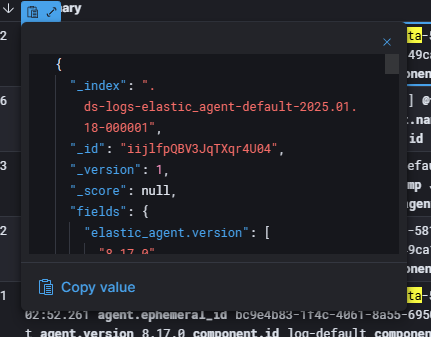
**LOGS – Elastic.**

Revisaremos los logs desde el apartado Discover > Search.

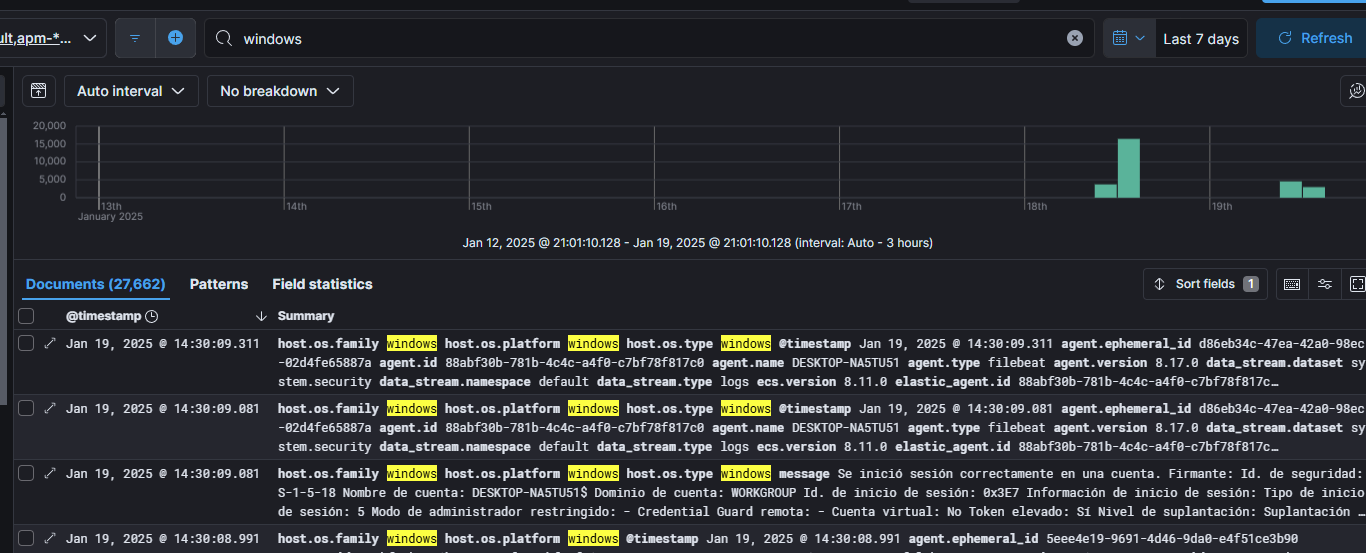
**Suricata:**



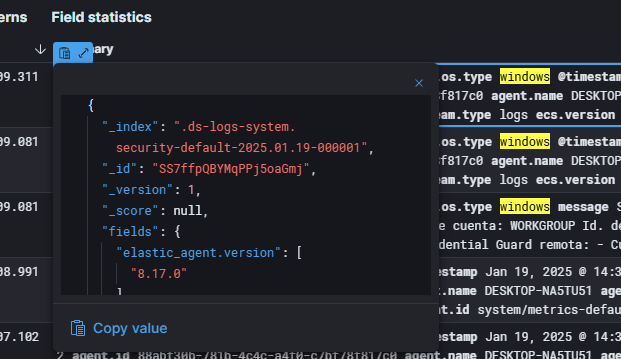
Copiaremos los valores desde Elastic y los incluimos en fiche “LogElastic\_Suricata.txt”



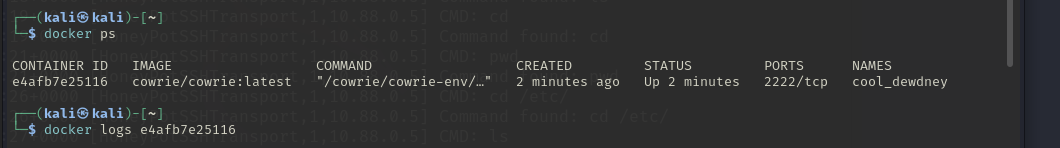
**Wind11:**

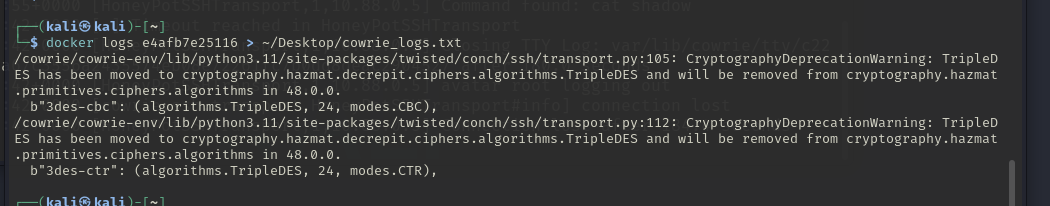


Copiaremos los valores desde Elastic y los incluimos en fiche “LogElastic\_Wind11.txt”



Considerando problemas en migrar los logs desde Kali a Elastic, comparto el archivo manual de los logs realizado en la máquina.





Adjunto el Archivo con el resto de la información.

**RESUMEN.**

Diagramamos una infraestructura para conectando 3 redes al UTM pfSense, entendiendo las limitaciones de una red doméstica.

Creamos el UTM desde la ISO (pfSense) simulando un entorno laboral, con la posibilidad de dirigir, redirigir, cancelar, bloquear y controlar las conexiones internas y externas.

Definimos las configuraciones para cada máquina según el requerimiento solicitado (conexiones y restricciones).

Asignamos reglas de Firewall para cada acceso y definimos una IP estática para conectarse desde una red externa.

Activamos y configuramos Elastic, definiendo integraciones para cada máquina virtual.

Enrollamos cada maquina con el server de Elastic para enviar, almacenar y analizar los logs de cada una.

**Herramientas Utilizadas**

* Virtual Box
* MV\_UTM-01
* MV\_Kali
* MV\_Wind11
* <https://www.elastic.co/es/cloud>
* GitHub

Ficheros Entregados

* Practica Blue Team.docx
* LogElastic\_suricata.txt
* LogElastic\_Wind11.txt
* LogCowrie.txt